



Υλοποίηση παράλληλης εφαρμογής Android για αναγνώριση χαρτονομισμάτων για άτομα με προβλήματα όρασης

**Banknotes recognition by parallel Android application for
people with visual impairment**



ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΩΝ

Διπλωματική εργασία
της
Ευρύκλειας Κεσίσογλου



Βόλος, Ιανουάριος 2016





Υλοποίηση παράλληλης εφαρμογής Android για αναγνώριση χαρτονομισμάτων για άτομα με προβλήματα όρασης

**Banknotes recognition by parallel Android application for
people with visual impairment**



Διπλωματική εργασία
της
Ευρύκλειας Κεσίσογλου

Επιβλέποντες:

Δημήτριος Κατσαρός
Επίκουρος Καθηγητής Π.Θ.

Παναγιώτης Μποζάνης
Καθηγητής Π.Θ.



Εγκρίθηκε από την διμελή εξεταστική επιτροπή την ημερομηνία εξέτασης.

(Υπογραφή)

(Υπογραφή)

.....

Δημήτριος Κατσαρός

Επίκουρος Καθηγητής Π.Θ.

.....

Παναγιώτης Μποζάνης

Καθηγητής Π.Θ.



(Υπογραφή)

.....

Ευρύκλεια Κεσίσογλου

Διπλωματούχος Μηχανικός Ηλεκτρονικών Υπολογιστών, Τηλεπικοινωνιών και
Δικτύων του Τμήματος Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών,
Πανεπιστημίου Θεσσαλίας

© 2016– All rights reserved



Βόλος, Ιανουάριος 2016



Στους γονείς μου, Βασίλη και Μαίρη



Ευχαριστίες

Μετά την περάτωση της εργασίας αυτής, θα ήθελα να ευχαριστήσω θερμά τους επιβλέποντες κ. Δημήτριο Κατσαρό και κ. Παναγιώτη Μποζάνη για την εμπιστοσύνη που μου έδειξαν, την άριστη συνεργασία και τις ουσιώδεις υποδείξεις και παρεμβάσεις που διευκόλυναν την εκπόνηση της διπλωματικής εργασίας.

Με την ευκαιρία αυτή θα ήθελα να ευχαριστήσω την οικογένεια μου για την υποστήριξη και την ενθάρρυνση που μου έδωσαν κατά τη διάρκεια των σπουδών μου και τους φίλους μου για τις όμορφες στιγμές που περάσαμε μαζί αυτά τα χρόνια.

Εύα Κεσίσογλου,
Βόλος 2016



Περίληψη

Σκοπός της παρούσας εργασίας είναι η ανάπτυξη και η σχεδίαση μιας εφαρμογής που θα τρέχει σε περιβάλλον Android και στόχο έχει να διευκολύνει τους χρήστες που έχουν προβλήματα όρασης στις καθημερινές τους χρηματικές συναλλαγές.

Συγκεκριμένα, οι χρήστες που έχουν κάποια αναπηρία στην όραση θα μπορούν να χρησιμοποιήσουν την εφαρμογή αυτή ώστε να αναγνωρίζουν τα χαρτονομίσματα που ανταλλάσσουν σε καθημερινές συναλλαγές με ασφάλεια. Η εφαρμογή αυτή θα είναι εύκολη στη χρήση της ώστε να διευκολύνει όσους αντιμετωπίζουν παρόμοια προβλήματα.



Abstract

The goal of this thesis is to develop an Android application that recognizes Euro Banknotes. This application is designed for people with vision impairment, so that it will make their daily transactions safe and easier. Image recognition is performed with Neural networks, a model employed in statistics and artificial intelligence.





Περιεχόμενα

1. Εισαγωγή.....	16
1.1 Σκοπός εργασίας.....	16
1.2 Κίνητρο.....	16
2. Θεωρητικό υπόβαθρο.....	17
2.1 Smartphones.....	17
2.2 Android.....	19
2.2.1 Android Market.....	21
2.2.2 Ιστορική αναδρομή.....	21
Εκδόσεις Android.....	22
2.2.3 Εφαρμογές Android.....	30
2.2.4 Αρχιτεκτονική Android.....	31
3 . Αναγνώριση εικόνας (Image recognition).....	34
3.1 Kohonen's Self Organizing Map.....	35
4. Νευρωνικά δίκτυα.....	37
4.1 Νευρώνας.....	40
Συναρτήσεις ενεργοποίησης	41
4.2 Μέθοδοι εκπαίδευσης νευρωνικών δικτύων.....	44
4.3 Perceptron (Αισθητήρας).....	46
Κανόνας Δέλτα (delta rule)	47
4.4 Πολυεπίπεδοι αισθητήρες (Multilayer Perceptrons, MLPs).....	48
4.5 Αλγόριθμος ανάστροφης διάδοσης (Back-propagation algorithm).....	49
5. Εργαλεία ανάπτυξης της εφαρμογής.....	52
5.1 Android Studio.....	52
5.2 Android SDK.....	53
5.3 Neuroph.....	54
5.4 Neuroph Studio.....	55



6. Υλοποίηση εφαρμογής.....	58
6.1 Αναγνώριση εικόνας με το Neuroph Studio.....	58
6.1.1 Δημιουργία νευρωνικού δικτύου.....	60
6.2 Λειτουργία εφαρμογής.....	68
6.3 Δομή της εφαρμογής.....	70
6.4 Απαιτήσεις εφαρμογής.....	78
 7. Συμπεράσματα.....	 79
7.1 Δοκιμές εφαρμογής.....	79
7.2 Εξέλιξη της εφαρμογής.....	80
 Ευρετήριο Εικόνων.....	 82
Ευρετήριο Πινάκων.....	83
Βιβλιογραφία.....	84
Σύνδεσμοι	85





1. Εισαγωγή

1.1 Σκοπός εργασίας

Η εργασία αυτή έχει ως σκοπό την ανάπτυξη εφαρμογής σε λειτουργικό σύστημα Android η οποία θα κάνει οπτική αναγνώριση των χαρτονομισμάτων με χρήση κάμερας και απευθύνεται κυρίως σε άτομα που έχουν προβλήματα στην όραση. Συγκεκριμένα, έχει ως σκοπό να διευκολύνει την καθημερινότητα όσων δυσκολεύονται να ξεχωρίσουν τα χαρτονομίσματα.

1.2 Κίνητρο

Αυτό που με οδήγησε στο να επιλέξω το συγκεκριμένο θέμα για την εργασία μου ήταν αρχικά η ανάγκη που υπάρχει σε ένα ποσοστό ανθρώπων με προβλήματα στην όραση να διευκολύνονται στις καθημερινές τους συναλλαγές. Σύμφωνα με έρευνα που έγινε από τον WHO (World Health Organization) τον Αύγουστο του 2014, 285 εκατομμύρια άνθρωποι σε όλο τον κόσμο έχουν πρόβλημα στην όραση. Από αυτούς τα 39 εκατομμύρια είναι τυφλοί ενώ 246 εκατομμύρια έχουν χαμηλή όραση.^[1] Το ποσοστό των ανθρώπων που αδυνατούν να αναγνωρίσουν από μόνοι τους τα χαρτονομίσματα είναι αρκετά υψηλό με αποτέλεσμα να πέφτει το επίπεδο της ποιότητας της ζωής τους καθώς δυσκολεύονται στις καθημερινές τους συναλλαγές. Με τη ραγδαία εξάπλωση και την αυξανόμενη χρήση των smartphones στην καθημερινότητα μας, είναι δυνατή η σχεδίαση μιας εφαρμογής αναγνώρισης χαρτονομισμάτων που θα έλυνε το παραπάνω πρόβλημα. Ένας ακόμη σημαντικός λόγος που επέλεξα το θέμα αυτό είναι ότι συνδυάζεται το ενδιαφέρον θέμα της εφαρμογής αυτής με έναν σημαντικό κλάδο της Τεχνητής νοημοσύνης: τα νευρωνικά δίκτυα.



2. Θεωρητικό υπόβαθρο

2.1 Smartphones

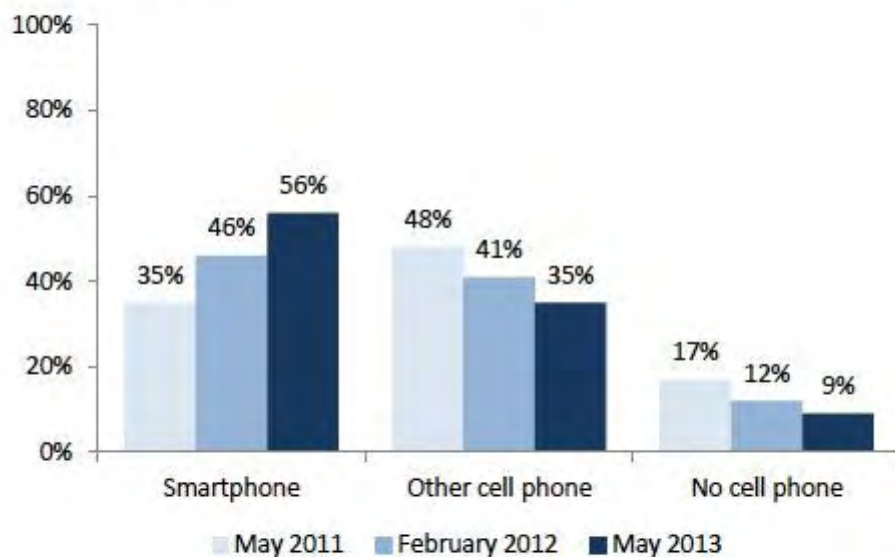
Τα Smartphones ή αλλιώς τα έξυπνα τηλέφωνα έχουν κατακλύσει την καθημερινότητα μας τα τελευταία χρόνια. Στην ουσία τα κινητά τηλέφωνα έχουν αναβαθμιστεί σε έξυπνα χάρη στο λειτουργικό τους σύστημα που τους προσφέρει περισσότερες δυνατότητες από ότι τα συμβατικά κινητά τηλέφωνα. Επίσης πέρα από το αναβαθμισμένο λειτουργικό τους σύστημα, έχουν ενσωματωμένες κάμερες υψηλής ανάλυσης, αισθητήρες κίνησης κ.α., GPS και οθόνες αφής. Οι νέες αυτές δυνατότητες αποτέλεσαν το λόγο της ραγδαίας εξάπλωσης τους τα τελευταία χρόνια καθώς πλέον αποτελούν αναπόσπαστο πολυεργαλείο της καθημερινότητας μας.^[2]

Σύμφωνα με έρευνα που έγινε το 2013, τα συμβατικά κινητά αντικαταστάθηκαν από έξυπνα τηλέφωνα με εκθετικό ρυθμό αν σκεφτεί κανείς πως τον Μάιο του 2011 το 35% του πληθυσμού ήτανε κάτοχος smartphone ενώ τον ίδιο μήνα του 2013 το ποσοστό αυτό ανέβηκε στο 56% όπως φαίνεται στην παρακάτω εικόνα.

Ένα ακόμα ποσοστό που παρουσιάζει ενδιαφέρον είναι αυτό των μη κατόχων κινητού τηλεφώνου που μειώθηκε ραγδαία από 17% στο 9%.

Changes in smartphone ownership, 2011–2013

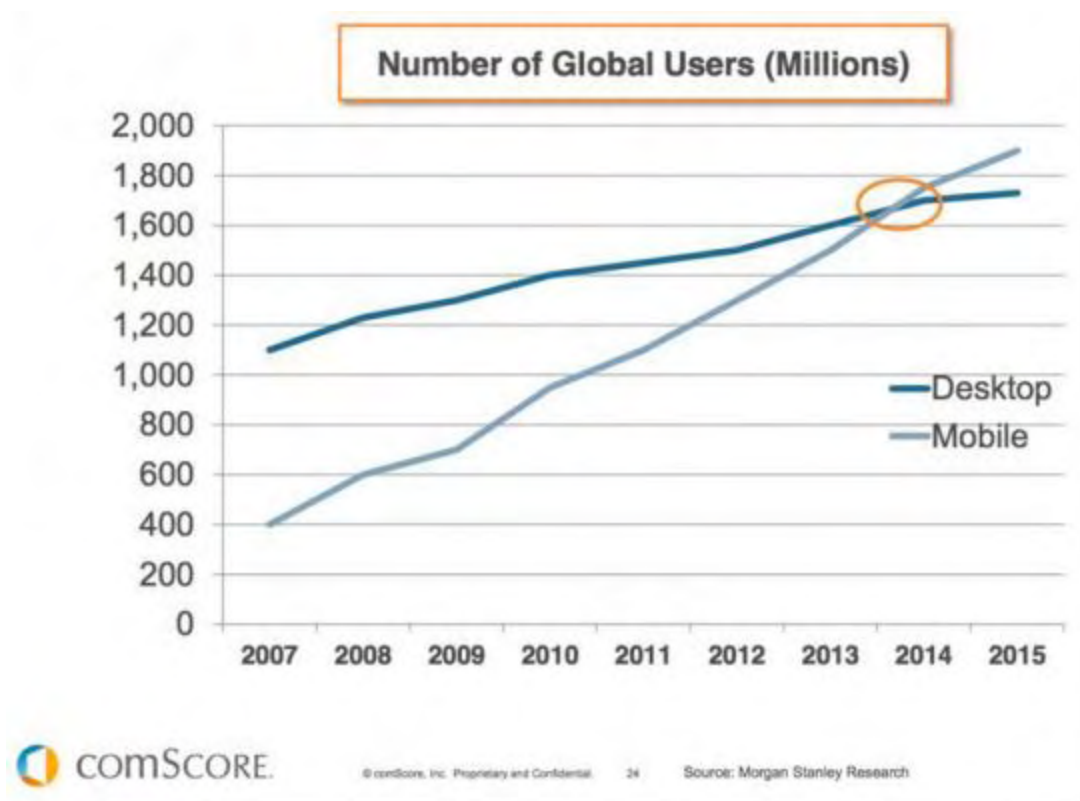
% of all U.S. adults who own...



Source: Pew Research Center's Internet & American Life Project April 26-May 22, 2011, January 20-February 19, 2012, and April 17-May 19, 2013 tracking surveys. For 2013 data, n=2,252 adults and survey includes 1,127 cell phone interviews. All surveys include Spanish-language interviews.

Εικόνα 2.1.1: Κατοχή έξυπνων τηλεφώνων

Στην παρακάτω εικόνα παρουσιάζονται τα αποτελέσματα διαφορετικής έρευνας που αφορά τον αριθμό χρηστών έξυπνων τηλεφώνων και ηλεκτρονικών υπολογιστών ανά τον κόσμο. Εκπληκτικό είναι το γεγονός ότι ο αριθμός των χρηστών έξυπνων τηλεφώνων έχουν πλέον υπερβεί αυτό των χρηστών σταθερού υπολογιστή στις αρχές του 2014, αγγίζοντας σχεδόν τον αριθμό των 2 δισεκατομμυρίων χρηστών.^[3]



Εικόνα 2.1.2: Χρήστες έξυπνων τηλεφώνων και ηλεκτρονικών υπολογιστών σε παγκόσμιο επίπεδο

2.2 Android

Το πιο δημοφιλές λειτουργικό σύστημα για συσκευές κινητής τηλεφωνίας είναι το Android, το οποίο τρέχει τον πυρήνα του λειτουργικού Linux. Αρχικά αναπτύχθηκε από την Google και αργότερα από την [Handset Alliance|Open Handset Alliance].

Η πρώτη έκδοση του λειτουργικού αυτού κυκλοφόρησε το Σεπτέμβριο του 2008 ενώ αναβαθμίζεται τακτικά μέχρι και σήμερα. Το Android είναι σχεδιασμένο ειδικά για συσκευές με οθόνη αφής και επιτρέπει στους προγραμματιστές εφαρμογών τη χρήση της γλώσσας προγραμματισμού Java, ελέγχοντας τη συσκευή μέσω βιβλιοθηκών λογισμικού της Google.



Εικόνα 2.2.1: Λογότυπο Android



Το Android είναι το πιο ευρέως διαδεδομένο λογισμικό στον κόσμο. Οι συσκευές με Android έχουν περισσότερες πωλήσεις από όλες τις συσκευές Windows, IOS και Mac OS X μαζί. Πολλές κατασκευαστικές εταιρίες κινητών τηλεφώνων πλέον χρησιμοποιούν το λειτουργικό Android για τα smartphones τους, μερικές από αυτές είναι η LG, Samsung, HTC, Lenovo, Sony Ericsson, Xiaomi κ.α. Ένα βασικό προνόμιο των συσκευών που τρέχουν το λογισμικό αυτό είναι πως είναι αφενός multimedia (παρέχεται δηλαδή η δυνατότητα να αναπαραχθούν πολλά μέσα) και multitasking (όπου δίνεται η δυνατότητα εκτέλεσης πολλών εφαρμογών ταυτόχρονα όπως για παράδειγμα να υποστηρίζεται η αναπαραγωγή μουσικής ταυτόχρονα με την αποστολή μηνύματος χωρίς να χρειάζεται να τερματιστεί κάποια από τις δύο). Η περιήγηση στο διαδίκτυο είναι ταχύτατη με δυνατότητα επιλογής browsers που υποστηρίζουν και flash. Ανεξάρτητα από το κόστος της συσκευής, όλες διαθέτουν GPS και Wi-Fi, δικαιώνοντας έτσι το βασικό λόγο δημιουργίας του εν λόγω λειτουργικού συστήματος που δεν είναι άλλος παρά η ανεμπόδιστη και εύκολη πρόσβαση στο διαδίκτυο, σε συνδυασμό με ένα πλήθος εφαρμογών (apps), όπως χάρτες, αναζήτηση, chat και e-mail, που πραγματικά επιτρέπουν στο χρήστη να μένει διαρκώς δικτυωμένος και ενημερωμένος.

Βασικό χαρακτηριστικό του Android, επίσης, είναι η πληθώρα εφαρμογών που διατηρούν τη συνεχή σύνδεση με Facebook, MySpace, Twitter και δεκάδες άλλες υπηρεσίες social networking. Ακόμη, το Android δίνει τη δυνατότητα να προσθήκης widget, δηλαδή εικονίδια για την ταχύτερη πρόσβαση στα προγράμματα, τα οποία τοποθετούνται στη home screen του κινητού (launcher). ^{[4][5]}

2.2.1 Android Market

Το Android Market, είναι ένα online κατάστημα της Google, προσφέρει σε κάθε χρήστη εφαρμογές για το κινητό τους που είναι συμβατό με το λειτουργικό της Google. Το συντριπτικό ποσοστό των εφαρμογών είναι δωρεάν ενώ πλέον οι πληρωμένες εφαρμογές είναι διαθέσιμες και στο ελληνικό κοινό.

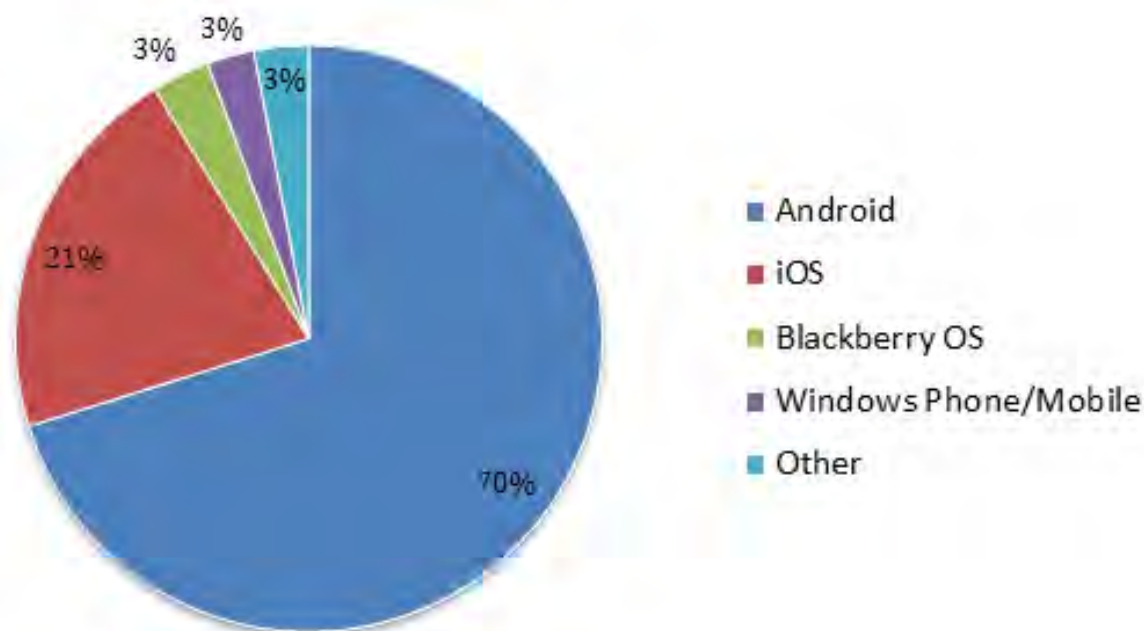


Εικόνα 2.2.1.1: Λογότυπο Android market

2.2.2 Ιστορική αναδρομή του Android

Το Android είναι ένα δωρεάν, ανοιχτού κώδικα λειτουργικό που κυκλοφόρησε δοκιμαστικά το Νοέμβριο του 2007 με την ονομασία Android beta, ενώ σχεδόν ένα χρόνο μετά τον Σεπτέμβριο του 2008 κυκλοφόρησε η πρώτη εμπορική έκδοση, το Android 1.0. Το λειτουργικό είναι υπό συνεχή ανάπτυξη από την Google και την Open Handset Alliance (OHA), και έχει πραγματοποιηθεί μια σειρά από ενημερώσεις στη λειτουργία του συστήματος από την αρχική κυκλοφορία του μέχρι και σήμερα.

Εκτός από το Android 1.0 και 1.1, όλες οι επόμενες εκδόσεις ακολούθησαν αλφαβητική σειρά και πήραν το όνομα τους από γλυκά: Cupcake, Donut, Ice Cream Sandwich, Kitkat κ.α. Το πρώτο κινητό που έτρεχε το λειτουργικό Android ήταν το T-Mobile G1 που κυκλοφόρησε στην αγορά το 2008. Τον Σεπτέμβρη του 2014 ανακοινώθηκε από τον αντιπρόεδρο της Google για το Android, τον Σουντάρ Πιτσάι πως το λειτουργικό σύστημα Android ξεπέρασε τις ένα δισεκατομμύριο ενεργοποιήσεις σε smartphones και tablets^[6]



Εικόνα 2.2.2.1: Το λειτουργικό Android αποτελεί το πιο διαδεδομένο λειτουργικό για κινητά.

Εκδόσεις Android

Android 1.5 Cupcake

API Level: 3

Η έκδοση αυτή παρουσιάστηκε τον Απρίλιο του 2009 βασισμένη στον Linux kernel 2.6.27. Είναι η πρώτη έκδοση που έχει όνομα επιδόρπιου, μια πρακτική που ακολουθήθηκε και στις επόμενες ονομασίες εκδόσεων. Στην έκδοση αυτή, νέες λειτουργίες της κάμερας υποστηρίζονται όπως είναι η καταγραφή και παρακολούθηση βίντεο από τη λειτουργία της κάμερας.



Εικόνα 2.2.2.2: Λογότυπο Android Cupcake

Επίσης δίνεται η δυνατότητα μεταφόρτωσης του βίντεο και των φωτογραφιών στο Youtube και το Picasa απευθείας από το τηλέφωνο. Μέσα στις νέες λειτουργίες βρίσκεται και το έξυπνο πληκτρολόγιο με πρόβλεψη κειμένου ενώ πλέον υποστηρίζει πρότυπο Bluetooth A2DP και AVRCP και συνδέεται αυτόματα σε μικροσυσκευές Bluetooth από συγκεκριμένη απόσταση. Στα χαρακτηριστικά αυτής της έκδοσης προστίθεται και το νέο γραφικό περιβάλλον με κινούμενες μεταβάσεις οθόνης. ^[13]

Android 1.6 Donut

API Level: 4

Η έκδοση Donut κυκλοφόρησε το Σεπτέμβριο του 2009 βασισμένη στον Linux kernel 2.6.29. Βελτιώθηκε η ταχύτητα απόκρισης και πλέον υποστηρίζεται η επιλογή πολλαπλών αρχείων ταυτόχρονα. Αναβαθμίστηκε ακόμα η φωτογραφική μηχανή, η συλλογή φωτογραφιών και το Android Market. Βελτιώθηκε η φωνητική αναζήτηση ώστε να έχει καλύτερη απόκριση και να ανταποκρίνεται καλύτερα στις ενσωματωμένες εφαρμογές, όπως είναι η κλήση επαφών. Παρέχεται ακόμη η δυνατότητα αναζήτησης σελιδοδεικτών, ιστορικού, επαφών από την αρχική οθόνη. Βελτίωση υποστήριξης τεχνολογιών CDMA/EVDO, 802.1x, VPNs κ.α. ^[14]



Εικόνα 2.2.2.3: Λογότυπο Android Donut



Android 2.0 Eclair

API level: 5

Η έκδοση αυτή έγινε διαθέσιμη τον Οκτώβρη του 2009. Επεκτάθηκε ο συγχρονισμός του λογαριασμού επιτρέποντας τους χρήστες να προσθέτουν περισσότερους λογαριασμούς για συγχρονισμό των mails και των επαφών. Βελτιώθηκαν οι υπηρεσίες κλήσης και αποστολής μηνυμάτων. Αναβαθμίστηκε σημαντικά η κάμερα, καθώς πλέον δίνεται η δυνατότητα χρήσης φλας, ψηφιακό zoom και άλλων λειτουργιών. Βελτίωση του εικονικού πληκτρολογίου και του ημερολογίου. Υποστήριξη του Bluetooth 2.1. ^[15]



*Εικόνα 2.2.2.4: Λογότυπο
Android Eclair*

Android 2.2–2.2.3 Froyo

API level: 8

Τον Μάιο του 2010 κυκλοφόρησε η έκδοση Froyo (συντομογραφία του frozen yogurt). Στην έκδοση αυτή υπήρξε σημαντική βελτίωση στην ταχύτητα, τη μνήμη και την απόδοση. Ενσωματώθηκε η V8 JavaScript engine του Chrome στον περιηγητή του Android. Επίσης, πλέον υποστηρίζονται οι υπηρεσίες Android Cloud to Device Messaging (C2DM), Adobe Flash. Προστέθηκε η δυνατότητα USB tethering και Wi-Fi hotspot. ^[16]



Εικόνα 2.2.2.5: Λογότυπο Android Froyo



Android 2.3 Gingerbread

API level: 9-10

Η έκδοση Gingerbread κυκλοφόρησε τον Δεκέμβριο του 2010. Περιείχε σημαντικές αναβαθμίσεις στο UI κάνοντας το πιο γρήγορο και εύχρηστο. Υποστηρίζονται μεγαλύτερες αναλύσεις οθόνης (WXGA και μεγαλύτερες). Προστέθηκαν νέα εφέ ήχου, όπως equalization, bass boost. Πλέον υποστηρίζονται παραπάνω από μια κάμερες, όπως μια μπροστινή κάμερα, εάν υπάρχει. Αναβαθμίστηκε το εικονικό πληκτρολόγιο, η δυνατότητα αντιγραφής-επικόλλησης καθώς και η διαχείριση ενέργειας. Τέλος, υποστηρίζεται η υπηρεσία Near Field Communication (NFC), επιτρέποντας στον χρήστη να διαβάσει μια NFC ετικέτα ενσωματωμένη σε ένα πόστερ, αυτοκόλλητο ή διαφήμιση.^[17]



Εικόνα 2.2.2.6: Λογότυπο Android Gingerbread

Android 3.0-3.2 Honeycomb

API level: 11-13

Η έκδοση αυτή έγινε διαθέσιμη τον Φεβρουάριο του 2011 και ήταν η πρώτη διαθέσιμη μόνο για tablet. Οι αναβαθμίσεις στην έκδοση αυτή αφορούν κυρίως την βελτίωση της υποστήριξης για tablets. Το νέο διαφορετικό UI που υποστηρίζει



διπύρηνους και τετραπύρηνους επεξεργαστές περιέχει την System Bar, για γρήγορη πρόσβαση σε ειδοποιήσεις στο κάτω μέρος της οθόνης καθώς και την Action Bar που



Εικόνα 2.2.2.7: Λογότυπο Android Honeycomb

δίνει πρόσβαση σε ρυθμίσεις, widget στο πάνω μέρος της οθόνης. Απλοποιήθηκε το multitasking καθώς πλέον είναι διαθέσιμα στιγμιότυπα των τρεχόντων εφαρμογών απλά με το πάτημα των Recent Applications στο System Bar. Υπάρχει ακόμη η δυνατότητα για Video Chat μέσω της εφαρμογής Google Talk και η ανάγνωση βιβλίων μέσω του Google eBooks. Τέλος, υποστηρίζεται πλέον κρυπτογράφηση των δεδομένων του χρήστη.^[18]

Android 4.0–4.0.2 Ice Cream Sandwich

API level: 14-15

Η έκδοση Ice Cream Sandwich ανακοινώθηκε στο κοινό τον Οκτώβρη 2011.

Στην έκδοση αυτή βελτιώθηκε σημαντικά η ταχύτητα και η απόδοση. Η μονάδα επεξεργασίας γραφικών (GPU) υποστηρίζει πλέον το OpenGL ES 2.0. Προστέθηκαν εικονικά πλήκτρα τα οποία μέχρι τώρα υπήρχαν στις συσκευές ως κανονικά πλήκτρα ή αφής. Δίνεται η δυνατότητα πρόσβασης σε εφαρμογές απευθείας από την οθόνη κλειδώματος. Βελτίωση της αντιγραφής-επικόλλησης, της δημιουργίας φακέλων και της περιήγησης στο ημερολόγιο. Αναβάθμιση της ασφάλειας του συστήματος με την προσθήκη αναγνώρισης προσώπου για το ξεκλείδωμα της συσκευής. Δίνεται η δυνατότητα



Εικόνα 2.2.2.8: Λογότυπο Android Ice Cream Sandwich

τερματισμού εφαρμογών που τρέχουν στο background και η επιλογή ορίου στην κίνηση δεδομένων. Επιτρέπεται πλέον η βιντεοσκοπήση σε ανάλυση 1080p. ^[19]

Android 4.1–4.3 Jelly Bean

API level: 16-18

Η έκδοση αυτή αναμενόταν να ανακοινωθεί σε εκδήλωση στη Νέα Υόρκη τον Οκτώβριο του 2012, κάτι που δεν έγινε τελικά λόγω του τυφώνα Σάντι. Αντί αυτού, κυκλοφόρησε με ανακοίνωση τύπου τον Νοέμβριο της ίδιας χρονιάς. Σημαντικές βελτιώσεις έγιναν στο κλείδωμα οθόνης και στη διαχείριση ενέργειας όταν το κινητό βρίσκεται σε αδράνεια.

Επιτρέπονται περισσότεροι λογαριασμοί χρηστών στην έκδοση που προορίζεται για tablets. Αναβαθμίζεται το

πληκτρολόγιο με σημαντικότερη βελτίωση την πρόβλεψη της επόμενης λέξης που εκτιμάται ότι θα γράψει ο χρήστης, με

βάση το ιστορικό συγγραφής του. Δίνεται επίσης η δυνατότητα χρήσης της υπηρεσίας Google Wallet καθώς και η χρήση εξωτερικής συσκευής USB ήχου. Στην επόμενη έκδοση 4.3 που ανακοινώθηκε τον Ιούλιο του 2013, αναβαθμίστηκαν τα γραφικά αφού πλέον υποστηρίζεται το OpenGL ES 3.0. Ακόμη υποστηρίζεται το Bluetooth Low Energy, νέα APIs για τους προγραμματιστές, καλύτερη πρόβλεψη γραφής σε όλα τα σημεία του λειτουργικού αλλά και βελτιώσεις στην δυνατότητα Multi-User. Μέσα στις βελτιώσεις του λογισμικού βρίσκονται και οι βελτιώσεις ταχύτητας και αυτονομίας για ακόμα καλύτερη εμπειρία χρήσης. ^[20]



Εικόνα 2.2.2.9: Λογότυπο Android Jelly Bean



Android 4.4–4.4.4 KitKat

API level: 19-20

Η έκδοση αυτή κυκλοφόρησε τον Σεπτέμβρη του 2013 και απαιτεί πλέον τουλάχιστον 340 MB RAM, ενώ όλες οι συσκευές με λιγότερη από 512 RAM θεωρούνται αργές. Αυτή η έκδοση, παρόλ' αυτά χρησιμοποιεί 16% λιγότερη μνήμη από ότι το Jelly Bean με αποτέλεσμα το λειτουργικό να είναι πιο ελαφρύ. Βελτιώθηκε σε μεγάλο βαθμό ο σχεδιασμός του λειτουργικού με αποτέλεσμα να είναι πιο εύχρηστο και ευέλικτο. Οι status και notification μπάρες είναι πλέον διαφανείς και δεν είναι ορατές μέσα από τις εφαρμογές. Αναβαθμίστηκε επίσης και η εφαρμογές κλήσεων, μηνυμάτων και φωνητικών αναζητήσεων. Ενσωματώθηκαν, ακόμη, υπηρεσίες cloud μέσα σε εφαρμογές, όπως είναι η photo gallery, GPS καθώς και υποστήριξη για αισθητήρες μέτρησης βημάτων.

[21]



Εικόνα 2.2.2.10: Λογότυπο Android KitKat

Android 5.0–5.1.1 Lollipop

API level 21-22

Η έκδοση αυτή ανακοινώθηκε τον Ιούνιο του 2014. Σημαντικές αναβαθμίσεις έγιναν άλλη μια φορά στη διαχείριση ενέργειας με την μπαταρία να παρατείνεται για 90 λεπτά με την επιλογή battery saver feature.



Βελτίωση της ασφάλειας της συσκευής με το kill switch, όπου αυτή μπλοκάρεται χωρίς τη γνώση κωδικού καθορισμένου από το χρήστη. Υπάρχει πλέον η δυνατότητα δημιουργίας πολλαπλών προφίλ ώστε να διευκολύνεται η χρήση για διαφορετικούς χρήστες. Υποστηρίζονται 64-bit επεξεργαστές και περισσότερες από μια sim. Βελτιώνονται τα γραφικά με αναβάθμιση σε OpenGL ES 3.1. Με την τελευταία αναβάθμιση του Lollipop, υποστηρίζονται και κλήσης ήχου υψηλής ανάλυσης.^[22]

*Εικόνα 2.2.2.11: Λογότυπο
Android Lollipop*

Android 6.0–6.0.1 Marshmallow

API level: 23

Η έκδοση Marshmallow παρουσιάστηκε ως Android M τον Μάιο του 2015. Εγκαινιάζεται ένα νέο σύστημα πληρωμών Android Pay, ενώ το Google Wallet δεν υποστηρίζεται πια.



*Εικόνα 2.2.2.12: Λογότυπο
Android Marshmallow*

Υποστηρίζεται η δυνατότητα αναγνώρισης δακτυλικών αποτυπωμάτων με αισθητήρες αφής, αναβαθμίζοντας την ασφάλεια της συσκευής. Νέο εμπλουτισμένο μενού στην επιλογή κειμένου και μηνύματα απαίτησης άδειας πρόσβασης σε δεδομένα από εφαρμογές. Σημαντική βελτίωση αποτελεί και η υιοθέτηση της εξωτερικής αποθήκευσης ώστε να λειτουργεί σαν την ενσωματωμένη. ^{[8] [23]}

2.2.3 Εφαρμογές Android



Εικόνα 2.2.3.1: Εφαρμογές

Το Android έχει μια μεγάλη κοινότητα προγραμματιστών που γράφουν εφαρμογές, οι οποίες επεκτείνουν τη λειτουργικότητα των συσκευών. Οι εφαρμογές γράφονται σε μια προσαρμοσμένη έκδοση της JAVA και μπορεί κανείς να κατεβάσει από το OnLine κατάστημα Google Play (πρώην Android Market) της Google όπως και από άλλα sites. Μέχρι τον Ιανουάριο του 2015 περισσότερες από 1.43 εκατομμύρια εφαρμογές ήταν διαθέσιμες για Android ενώ εκτιμάτε ότι ο αριθμός των downloads από το Android Market έχει υπερβεί τα 50 δισεκατομμύρια. Το Android είναι η πρώτη σε πωλήσεις παγκοσμίως πλατφόρμα για smartphone, καθώς μέχρι το Δεκέμβριο του 2014 μετρούσε περισσότερες από 1 δισεκατομμύριο συσκευές σε χρήση. ^{[7] [24]}



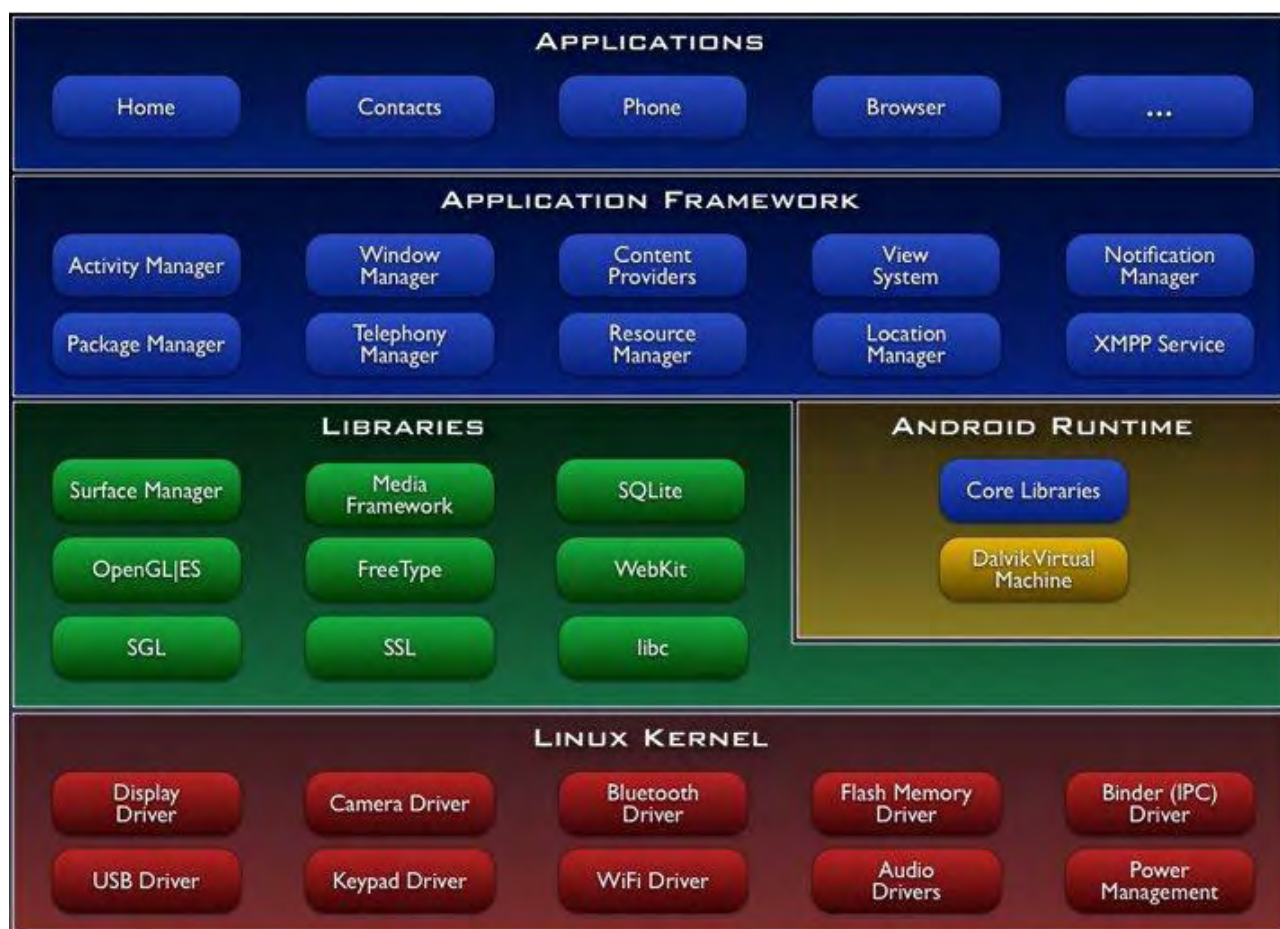
2.2.4 Αρχιτεκτονική Android

Η αρχιτεκτονική του Android περιλαμβάνει τα εξής επίπεδα, αρχίζοντας από το ψηλότερο και πηγαίνοντας στο χαμηλότερο:

Επίπεδο Εφαρμογών (Applications): Το Android είναι εξ αρχής εφοδιασμένο με ένα σύνολο από βασικές εφαρμογές που περιλαμβάνουν ένα email client, ένα πρόγραμμα για SMS μηνύματα, ημερολόγιο, χάρτες (Google Maps), περιηγητή ιστού, πρόγραμμα για δομημένη αποθήκευση των επαφών και άλλα. Επίσης, αναρίθμητες εφαρμογές είναι διαθέσιμες στο Android market. Όλες οι εφαρμογές είναι γραμμένες στην γλώσσα προγραμματισμού Java.

Επίπεδο Πλαισίου Εφαρμογών (Applications Framework): Παρέχοντας μια ανοικτή πλατφόρμα ανάπτυξης, το Android προσφέρει στους προγραμματιστές την δυνατότητα να κατασκευάσουν πλούσιες σε δυνατότητες και καινοτόμες εφαρμογές. Δίνεται η δυνατότητα στους προγραμματιστές να εκμεταλλευτούν πλήρως το υλικό της συσκευής, να έχουν πρόσβαση σε υπηρεσίες εντοπισμού θέσης, να τρέξουν υπηρεσίες στο background, να θέσουν χρονοδιακόπτες για εμφάνιση ειδοποιήσεων και πολλά άλλα. Επίσης έχουν πλήρη πρόσβαση στο ίδιο πλαίσιο από APIs που έχουν οι βασικές εφαρμογές του Android. Η αρχιτεκτονική είναι διαμορφωμένη με τέτοιο τρόπο που κάθε εφαρμογή μπορεί να χρησιμοποιήσει τις δυνατότητες μιας άλλης και επίσης με τρόπο που δίνει την δυνατότητα στον χρήστη να αλλάξει τα συστατικά κάθε εφαρμογής. Κάτω από το πλαίσιο των εφαρμογών υπάρχει ένα σύστημα από υπηρεσίες και συστήματα τα οποία περιλαμβάνουν:

- Ένα σύνολο από γραφικά στοιχεία (Views) για την δημιουργία γραφικού περιβάλλοντος συμπεριλαμβανομένων λιστών (lists), πλεγμάτων (grids), κουτιών κειμένου (text boxes), κουμπιών (buttons) και άλλων.
- Ένα διαχειριστή περιεχομένου (Content Manager) ο οποίος επιτρέπει στις εφαρμογές την πρόσβαση σε δεδομένα άλλων εφαρμογών ή τον διαμοιρασμό των δικών τους δεδομένων με άλλες εφαρμογές.
- Ένα διαχειριστή πόρων (Resource Manager) για την πρόσβαση στους πόρους όπως strings, εικόνες, layout files.
- Έναν διαχειριστή ειδοποιήσεων (Notification Manager) ο οποίος επιτρέπει την προβολή ειδοποιήσεων στην μπάρα κατάστασης (status bar).
- Έναν διαχειριστή δραστηριοτήτων (Activity Manager) ο οποίος διαχειρίζεται τον κύκλο ζωής των εφαρμογών.



Εικόνα 2.2.4.1: Αρχιτεκτονική Android

Επίπεδο Βιβλιοθηκών (Libraries): Το οποίο περιλαμβάνει ένα σύνολο από βιβλιοθήκες γραμμένες σε C/C++ (όπως είναι η SQLite για αποθήκευση και διαμοιρασμό δεδομένων, η OpenGL για 2D και 3D γραφικά κ.α.) οι οποίες χρησιμοποιούνται από διάφορα στοιχεία του συστήματος του Android. Οι δυνατότητες που προσφέρουν αυτές οι βιβλιοθήκες είναι προσβάσιμες στους προγραμματιστές δια μέσου του Applications Framework.



Επίπεδο Εκτέλεσης (Android Runtime): Το οποίο αποτελείται από ένα σύνολο από βασικές βιβλιοθήκες και την Dalvik Virtual Machine. Η Dalvik Virtual Machine είναι βασικό συστατικό της Android αρχιτεκτονικής που μοιάζει με την Java Virtual Machine με τη διαφορά πως είναι βελτιστοποιημένη για Android εφαρμογές. Είναι υπεύθυνη για διαχείριση μνήμης, multithreading και άλλες λειτουργίες.

Πυρήνας του Linux: Το Android βασίζεται στον πυρήνα Linux έκδοση 3.10 για βασικές υπηρεσίες συστήματος όπως ασφάλεια, διαχείριση μνήμης, διαχείριση διεργασιών, στοίβα δικτύου, και οδηγούς συσκευών. Ο πυρήνας λειτουργεί επίσης ως ένα ενδιάμεσο επίπεδο αφαίρεσης μεταξύ της στοίβας λογισμικού και του υλικού (κάμερα, πληκτρολόγιο, οθόνη κ.τλ.).^[9]



3 . Αναγνώριση εικόνας (Image recognition)

Η Όραση υπολογιστών αποτελεί κλάδο της μηχανικής υπολογιστών που ασχολείται με μεθόδους απόκτησης, ανάλυσης, επεξεργασίας και κατανόησης εικόνων και γενικά πολυδιάστατων δεδομένων στον πραγματικό κόσμο με σκοπό να παράγει αριθμητική ή συμβολική πληροφορία. Μια σημαντική κατηγορία, η οποία περιέχεται σε αυτόν τον κλάδο, είναι και η αναγνώριση εικόνας, με την οποία ένας οποιοσδήποτε υπολογιστής θα είναι σε θέση να αναγνωρίσει το περιεχόμενο της εικόνας μιμούμενος τις ικανότητες της ανθρώπινης όρασης.

Το κλασικό πρόβλημα της Όρασης υπολογιστών είναι το κατά πόσο ή όχι τα δεδομένα της εικόνας περιέχουν ένα συγκεκριμένο αντικείμενο, ένα χαρακτηριστικό ή μια δραστηριότητα. Οι πιο πετυχημένοι αλγόριθμοι για παρόμοια προβλήματα είναι βασισμένοι στα Νευρωνικά Δίκτυα. Στην ουσία, οι αλγόριθμοι αυτοί, εξάγουν ένα σύνολο χαρακτηριστικών της εικόνας βάση του οποίου ζητείται να ληφθεί η απόφαση για την ύπαρξη ενός αντικειμένου.

Στην παρούσα πτυχιακή εργασία γίνεται μια προσπάθεια εκπαίδευσης τεχνητού νευρωνικού δικτύου με επίβλεψη, έτσι ώστε να επιτευχθεί η επιτυχής αναγνώριση χαρτονομισμάτων για άτομα με προβλήματα οράσεως. Οι αλγόριθμοι που μελετήθηκαν είναι κυρίως βασισμένοι στα νευρωνικά δίκτυα. Μερικοί από αυτούς είναι ο αλγόριθμος της ανάστροφης διάδοσης (Back-propagation algorithm) και το νευρωνικό δίκτυο τύπου Kohonen (self-organizing map-SOM).

Οι περισσότερες τεχνικές εκπαίδευσης που χρησιμοποιούνται ευρέως στα νευρωνικά δίκτυα αφορούν αλγορίθμους όπως είναι ο back-propagation, όπου είναι απαραίτητοι δύο πίνακες: αυτός της εισόδου και ο πίνακας στόχων. Με τη συγκεκριμένη μέθοδο, τα δεδομένα παρουσιάζονται στο δίκτυο και η έξοδος αυτού συγκρίνεται με τον πίνακα στόχων. Αν αυτοί διαφέρουν, τότε τα βάρη του νευρωνικού δικτύου τροποποιούνται ώστε να μειωθεί το λάθος στην έξοδο. Η διαδικασία αυτή επαναλαμβάνεται αρκετές φορές μέχρις ότου το δίκτυο δώσει την επιθυμητή έξοδο. Ο αλγόριθμος ανάστροφης διάδοσης, ο αλγόριθμος που χρησιμοποιήθηκε για την εργασία αυτή, αναλύεται περαιτέρω στο επόμενο κεφάλαιο. ^[25]

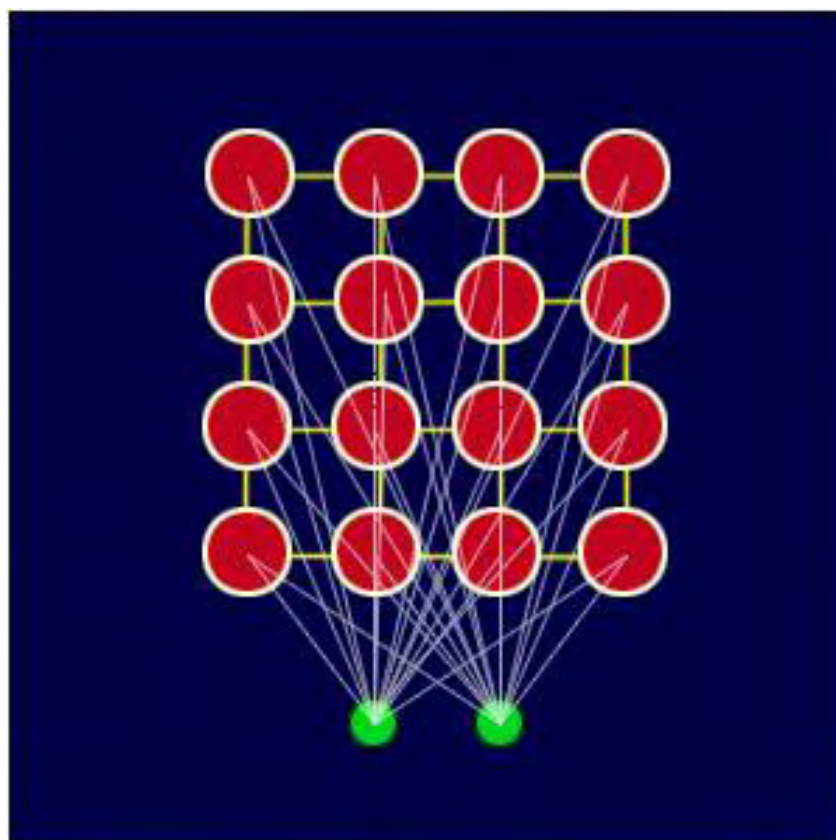


3.1 Kohonen's Self Organizing Map

Το μοντέλο αυτό σχεδιάστηκε από τον Τεοντο Kohonen και είναι ένας τύπος Τεχνητού Νευρωνικού Δικτύου που εκπαιδεύεται χωρίς επίβλεψη. Το μοντέλο αυτό επεξεργάζεται την είσοδο που θα του δοθεί και παράγει ένα συνήθως δισδιάστατο πίνακα αναπαράστασης της εισόδου, ταξινομημένο. Η φιλοσοφία του αλγορίθμου αυτού διαφέρει από τα κλασικά νευρωνικά δίκτυα καθώς βασίζεται στην ανταγωνιστική μάθηση σε αντίθεση με τα συστήματα στα οποία εφαρμόζεται η τεχνική της διόρθωσης με βάση το ποσοστό λάθους όπως γίνεται στον αλγόριθμο back-propagation. Μια από τις πιο ενδιαφέρουσες πτυχές των SOMs είναι πως μαθαίνουν να κατατάσσουν τα δεδομένα χωρίς επίβλεψη. Ο τρόπος με τον οποίο επεξεργάζονται τα δεδομένα που δέχονται ως είσοδο τα καθιστά πολύ χρήσιμα στην αναπαράσταση δεδομένων πολλών διαστάσεων σε λιγότερες.

Οι διεργασίες που εκτελούν οι SOMs είναι δύο. Η μία αφορά την εκπαίδευση χωρίς επίβλεψη, όπου ο χάρτης αυτό-οργανώνεται και η άλλη τον προσδιορισμό της θέσης ενός νέου στοιχείου εισόδου στον ήδη οργανωμένο χάρτη.

Ας υποθέσουμε ένα απλό παράδειγμα SOM, με δισδιάστατο πίνακα από κόμβους που είναι πλήρως συνδεδεμένο με το επίπεδο εισόδου όπως φαίνεται στην παρακάτω φωτογραφία.



Εικόνα 3.1.1: Παράδειγμα αρχιτεκτονικής SOM

Κάθε κόμβος έχει μια συγκεκριμένη τοπολογική θέση στο πλέγμα x-y και περιέχει ένα πίνακα βαρών της ίδιας διάστασης με τους πίνακες εισόδου. Όπου τα βάρη των κόμβων μοιάζουν με τα βάρη της εισόδου, η περιοχή του πλέγματος βελτιστοποιείται ώστε να αντιπροσωπεύει καλύτερα τα δεδομένα για την κλάση στην οποία ανήκει ο πίνακας εισόδου. Ενώ αρχικά είναι τυχαία αρχικοποιημένα τα βάρη των κόμβων, η διαδικασία αυτή επαναλαμβάνεται μέχρι ο SOM να οργανωθεί σε σταθερές κλάσεις. Η μέθοδος αυτή, αν και αποτελεσματική σε περιπτώσεις όπου απαιτείται κατηγοριοποίηση ή ομαδοποίηση των δεδομένων που δίνονται στην είσοδο, δεν ενδείκνυται σε περιπτώσεις αναγνώρισης εικόνας. Ο αλγόριθμος αυτός είναι σε θέση να κατηγοριοποιήσει τις εικόνες χωρίς να μπορεί όμως εύκολα να τις αναγνωρίσει. Για το λόγο αυτό, επιλέχθηκε το μοντέλο του νευρωνικού δικτύου με πολυεπίπεδους αισθητήρες και τη μέθοδο back-propagation για την παρούσα εργασία. ^{[26][27]}



4. Νευρωνικά δίκτυα

Τα τεχνητά νευρωνικά δίκτυα αποτελούν την ανθρώπινη απόπειρα προσομοίωσης του ανθρώπινου εγκεφάλου. Η αρχιτεκτονική τους βασίζεται στην αρχιτεκτονική των βιολογικών νευρωνικών δικτύων. Η δομή του εγκεφάλου επιτρέπει την παράλληλη επεξεργασία δεδομένων αλλά παρέχει και την δυνατότητα συνεχούς μάθησης μέσω της αλληλεπίδρασης με το περιβάλλον. Με αυτόν τον τρόπο επιτυγχάνεται η εκτέλεση δύσκολων καθηκόντων όπως είναι η αναγνώριση σχημάτων και άλλες λειτουργίες.



Εικόνα 4.1: Νευρωνικά δίκτυα, προσομείωση της λειτουργίας του ανθρώπινου εγκεφάλου

Ο ανθρώπινος εγκέφαλος αποτελείται από δισεκατομμύρια νευρικά κύτταρα, τους νευρώνες. Ο καθένας από αυτούς συνδέεται με χιλιάδες άλλους νευρώνες με συνδέσεις που ονομάζονται συνάψεις και επικοινωνεί με αυτούς μέσω ηλεκτροχημικών σημάτων. Ένας αριθμός νευρώνων μαζί με τις συνάψεις τους αποτελούν ένα βιολογικό νευρωνικό δίκτυο.

Το δομικό μοντέλο αυτό του εγκεφάλου που είναι σε θέση να επεξεργαστεί τόνους δεδομένων και να εκτελέσει δύσκολες λειτουργίες υιοθετήθηκε ως ένα χρήσιμο υπολογιστικό εργαλείο με πολλές εφαρμογές. Το πρώτο μοντέλο νευρωνικού δικτύου παρουσιάστηκε το 1943 από τους McCulloch και Pitts. Έκτοτε, παρουσιάστηκαν περισσότερα και πιο βελτιωμένα μαθηματικά μοντέλα τα οποία εφαρμόστηκαν σε αρκετούς τομείς όπως την Αεροδιαστημική (Αυτόματος πιλότος, Ανίχνευση βλαβών), Ηλεκτρονική (Τεχνητή όραση, Σύνθεση ήχου), Ιατρική (Ανάλυση συμπτωμάτων και διάγνωση για διάφορες ασθένειες) κ.α.

Τα τεχνητά νευρωνικά δίκτυα (ΤΝΔ), ως προσομοίωση του ανθρώπινου εγκεφάλου,



αποκτούν τη γνώση μέσα από τη διαδικασία της μάθησης. Οι δυνάμεις σύνδεσης των νευρώνων σαν συναπτικά βάρη, χρησιμοποιούνται για την αποθήκευση γνώσης. Η δομή ενός ΤΝΔ μοντελοποιήθηκε με βάση το βιολογικό νευρωνικό δίκτυο ώστε να εμφανίζει παρόμοιες δυνατότητες. Ένα τεχνητό νευρωνικό δίκτυο λοιπόν αποτελείται από ένα σύνολο κόμβων (ή αλλιώς νευρώνων) που αλληλεπιδρούν, συνδεδεμένοι μεταξύ τους μέσω των συνάψεων. Ο βαθμός αλληλεπίδρασης είναι διαφορετικός για κάθε ζεύγος νευρώνων αφού καθορίζεται από το ανάλογο βάρος.

Η λειτουργία ενός τεχνητού νευρωνικού δικτύου βασίζεται στην αλληλεπίδραση με το περιβάλλον και τη συνεχή μάθηση από αυτό. Έτσι, τα βάρη των συνάψεων μεταβάλλονται συνεχώς, αναβαθμίζοντας ή υποβαθμίζοντας την σημαντικότητα του κάθε δεσμού. Τα βάρη αυτά είναι προσδίδουν και τη δυνατότητα στο νευρωνικό δίκτυο να αποκτήσει εμπειρική γνώση και να προσαρμόζεται ανάλογα με το περιβάλλον.

Λόγω της δομής τους, τα νευρωνικά δίκτυα δεν λειτουργούν σειριακά, καθώς μια εργασία μοιράζεται στους διάφορους κόμβους του δικτύου, τους νευρώνες. Για τον λόγο αυτό ονομάζονται συστήματα παράλληλης κατανεμημένης επεξεργασίας. Αυτό έχει σαν αποτέλεσμα να πετυχαίνουν μεγάλες ταχύτητες κάτι που τα κάνει ευρέως δημοφιλή σε διάφορες εφαρμογές στην επιστήμη.

Δομικά, τα ΤΝΝ αποτελούνται από επίπεδα ή αλλιώς στοιβάδες που περιέχουν νευρώνες και συνδέονται μεταξύ τους με κάποια βάρη. Υπάρχουν τρία είδη επιπέδων που εκτελούν διαφορετικές λειτουργίες:

- Επίπεδο εισόδου (input layer)
- Κρυφό επίπεδο (hidden layer)
- Επίπεδο εξόδου (output layer)

Ο τρόπος με τον οποίον είναι συνδεδεμένα τα επίπεδα, ο αριθμός των κρυφών επιπέδων και ο τύπος των νευρώνων που έχει το κάθε επίπεδο καθορίζουν την αρχιτεκτονική του ΤΝΝ. Η επιλογή του αλγορίθμου εκμάθησης έχει άμεση σχέση με τη δομή ενός ΤΝΝ.



Τα νευρωνικά δίκτυα χωρίζονται σε δύο κύριες κατηγορίες:

➤ **Δίκτυα εμπρόσθιας τροφοδότησης (FeedForward)**

Η δομή αυτών των δικτύων αποτελείται από επίπεδα. Η ροή της πληροφορίας ξεκινά από το επίπεδο εισόδου (input layer) και καταλήγει πάντα στο επίπεδο εξόδου (output layer). Οι νευρώνες εισόδου μεταφέρουν την πληροφορία ενώ οι νευρώνες του επιπέδου εξόδου είναι υπολογιστικοί νευρώνες. Τέτοια δίκτυα είναι τα μονής στοιβάδας (single layer), πολλαπλών στοιβάδων (multiple layer) και ακτινικών συναρτήσεων βάσης (RBF).

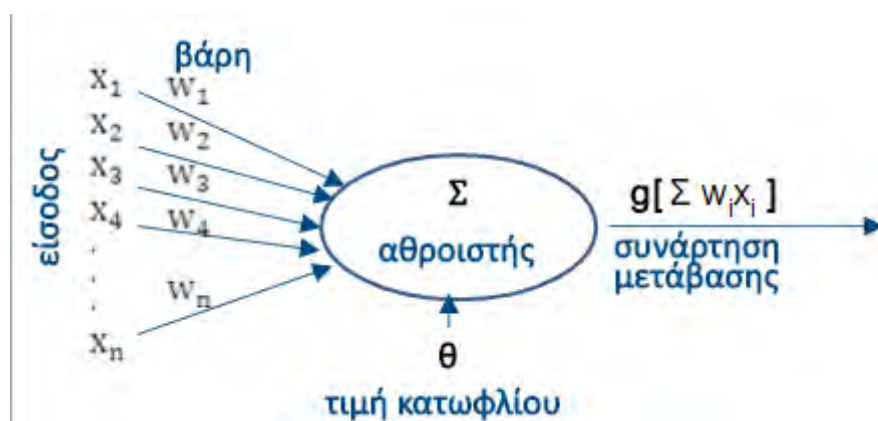
➤ **Επανατροφοδοτούμενα δίκτυα (Recurrent ή Feedback)**

Παράδειγμα τέτοιων δικτύων είναι τα δίκτυα Kohonen (self-organized ANNs), δίκτυα Hopfield, ανταγωνιστικά δίκτυα κ.α. ^{[28] [29]}

4.1 Νευρώνας

Ένας νευρώνας, στα ΤΝΔ, είναι μια μονάδα επεξεργασίας της πληροφορίας. Τα τρία βασικά στοιχεία αυτού του μοντέλου είναι:

- Ένα σύνολο από συνάψεις ή συνδετικούς κρίκους
- Ένας αθροιστής
- Μία συνάρτηση ενεργοποίησης



Εικόνα 4.1.1: Δομή και λειτουργία ενός νευρώνα

Ένας τεχνητός νευρώνας δέχεται ως είσοδο τα σήματα εισόδου x_i όπου $i=1..n$, στα οποία αντιστοιχεί μία τιμή βάρους w_i από την αντίστοιχη σύναψη. Η τιμή αυτή μπορεί να είναι θετική ή αρνητική αναλόγως την λειτουργία της σύναψης. Στη συνέχεια, ο αθροιστής δέχεται σαν είσοδο τα σήματα επηρεασμένα από την αντίστοιχη τιμή βάρους και υπολογίζει το άθροισμα $\sum w_i x_i$. Κατόπιν, το άθροισμα αυτό δέχεται σαν είσοδο η εκάστοτε συνάρτηση ενεργοποίησης ή συνάρτηση μετάβασης g .^[30]



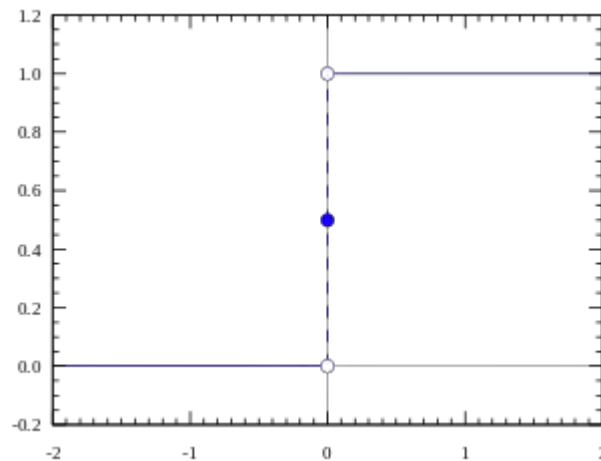
Συναρτήσεις ενεργοποίησης

Υπάρχουν πολλές κατηγορίες συναρτήσεων ενεργοποίησης όπως είναι οι βηματικές, γραμμικές, μη γραμμικές και στοχαστικές συναρτήσεις.

Η πιο απλή είναι η βηματική συνάρτηση ενεργοποίησης όπου έχει τη μορφή:

$$\phi(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

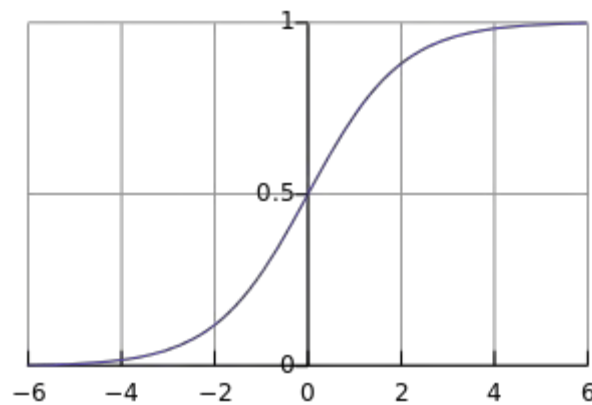
Η **βηματική συνάρτηση** δεν συνηθίζεται να χρησιμοποιείται στα τεχνητά νευρωνικά δίκτυα καθώς η παράγωγος της είναι ίση με το μηδέν. Η ανάγκη συναρτήσεων ενεργοποίησης που η γραφική τους παράσταση να μοιάζει με τη βηματική όμως να είναι συνεχής και παραγωγίσιμη σε όλο το πεδίο ορισμού οδήγησε στην σιγμοειδή, η οποία χρησιμοποιήθηκε και στην εφαρμογή αυτή. ^{[31][32]}



Εικόνα 4.1.2: Γραφική παράσταση βηματικής συνάρτησης

Η **σιγμοειδής συνάρτηση** είναι μια μαθηματική συνάρτηση η οποία έχει μορφή S και ονομάζεται επίσης και ως *σιγμοειδής καμπύλη*. Συχνά ως *σιγμοειδής συνάρτηση* αναφέρεται η ειδική περίπτωση της λογιστικής παλινδρόμησης και συγκεκριμένα ο παρακάτω μαθηματικός τύπος: ^[33]

$$S(t) = \frac{1}{1 + e^{-t}}$$

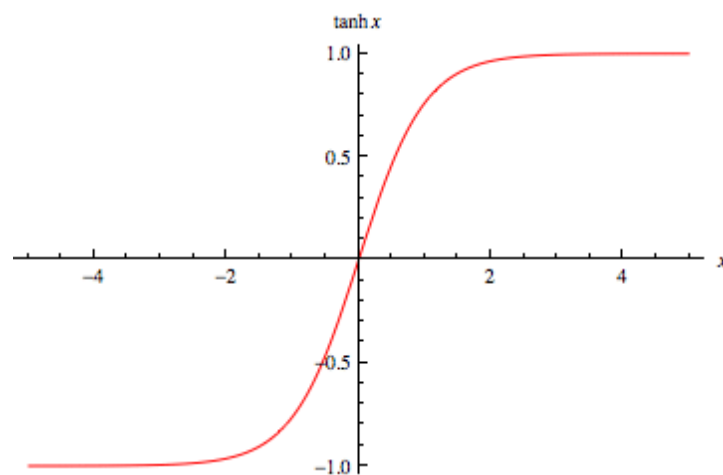


Εικόνα 4.1.3: Γραφική παράσταση σιγμοειδούς συνάρτησης

Υπερβολική εφαπτομένη (hyperbolic tangent - tanh)

Η υπερβολική εφαπτομένη είναι επίσης μια σιγμοειδής μη γραμμική συνάρτηση συχνά χρησιμοποιούμενη ως συνάρτηση ενεργοποίησης σε νευρωνικά δίκτυα.

$$\varphi(x) = \tanh(x) = \frac{1 - e^{-x}}{1 + e^{-x}}$$



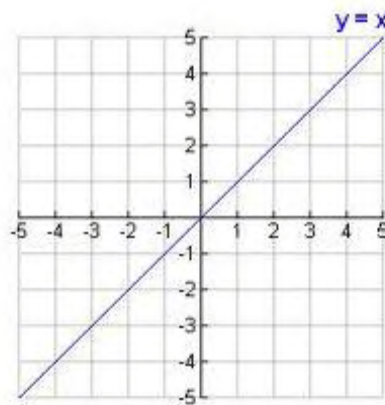
Εικόνα 4.1.4: Γραφική παράσταση υπερβολικής εφαπτομένης (tanh)



Γραμμική συνάρτηση (Linear function)

Η γραμμική συνάρτηση μπορεί να είναι η εξής ή οποιαδήποτε άλλη γραμμική ή κατά τμήματα γραμμική συνάρτηση^[36]:

$$f(x) = x$$



Εικόνα 4.1.5: Γραφική παράσταση γραμμικής συνάρτησης

4.2 Μέθοδοι εκπαίδευσης νευρωνικών δικτύων

Μια από τις πιο σημαντικές ιδιότητες των Νευρωνικών Δικτύων είναι η ικανότητα τους για εκπαίδευση. Η τελευταία επιτυγχάνεται με τη συνεχή ανταλλαγή τιμών και βαρών που αποσκοπεί στη βαθμιαία σύλληψη της πληροφορίας, η οποία θα είναι στη συνέχεια διαθέσιμη για ανάκτηση. Οι αλγόριθμοι που στόχο έχουν την προσαρμογή των τιμών των βαρών ενός τεχνητού νευρωνικού δικτύου είναι πολλοί και κατατάσσονται σε δυο κύριες κατηγορίες:



Μάθηση με επίβλεψη

Η μάθηση αυτή είναι μια διαδικασία η οποία συνδυάζει έναν εξωτερικό εκπαιδευτή και τη συνολική ή γενικευμένη πληροφορία. Κάποιες από τις μεθόδους οι οποίες συγκαταλέγονται σε αυτή την κατηγορία είναι η μάθηση με διόρθωση σφάλματος, η στοχαστική μάθηση. Παραδείγματα τα οποία αντιπροσωπεύουν την μάθηση με επίβλεψη συμπεριλαμβάνουν αποφάσεις για το πότε θα πρέπει να σταματήσει η διαδικασία εκπαίδευσης, αποφάσεις αναφορικά με τη συχνότητα παρουσίασης στο δίκτυο τα πρότυπα εκπαίδευσης και η παρουσίαση προόδου του δικτύου. Η μάθηση με επίβλεψη χωρίζεται σε δύο ακόμα κατηγορίες: στη δομική (structural) και στην προσωρινή (temporal) εκμάθηση.

Οι αλγόριθμοι οι οποίοι βρίσκονται στην πρώτη κατηγορία, χρησιμοποιούνται για την εύρεση της βέλτιστης σχέσης μεταξύ εισόδων και εξόδων για κάθε ξεχωριστό ζευγάρι προτύπων. Παραδείγματα της δομικής εκμάθησης αποτελούν η αναγνώριση και η κατηγοριοποίηση προτύπων, ενώ παραδείγματα της προσωρινής εκμάθησης η πρόβλεψη και ο έλεγχος.

Μάθηση χωρίς επίβλεψη

Οι αλγόριθμοι της εν λόγω μάθησης αναφέρονται ως αυτό-οργανωμένοι (self-organized) και είναι διαδικασίες οι οποίες δεν απαιτούν να είναι παρών ένας «εξωτερικός» δάσκαλος ή επιβλέπων. Βασίζονται, μάλιστα, μόνο σε τοπική πληροφορία καθ' όλη τη διάρκεια της εκπαίδευσης του Τεχνητού Νευρωνικού Δικτύου. Οι συγκεκριμένοι αλγόριθμοι οργανώνουν τα δεδομένα και ανακαλύπτουν τις σημαντικές συλλογικές ιδιότητες. Για παράδειγμα, αλγόριθμοι εκπαίδευσης χωρίς επίβλεψη είναι ο αλγόριθμος Hebbian, ο διαφορικός αλγόριθμος Hebbian και ο Min-Max αλγόριθμος.

Κατά κύριο λόγο οι περισσότερες διαδικασίες εκπαίδευσης είναι off line. Όταν χρησιμοποιείται όλο το δείγμα προτύπων για την τροποποίηση των τιμών των βαρών, πριν την τελική χρήση του δικτύου ως εφαρμογή, τότε ονομάζεται off line εκπαίδευση. Οι αλγόριθμοι εκπαίδευσης off line έχουν την απαίτηση να βρίσκονται στην εκπαίδευση του δικτύου παρόντα όλα τα πρότυπα. Το γεγονός αυτό αποκλείει την πιθανότητα εισαγωγής νέων πληροφοριών μέσω νέων προτύπων. Βέβαια, υπάρχουν και Τεχνητά Νευρωνικά Δίκτυα τα οποία δεν αποκλείουν την εισαγωγή νέας πληροφορίας, μετά την τελική τους μοντελοποίηση. Αν παρουσιαστεί ανάγκη εισαγωγής νέου προτύπου στο δίκτυο, μπορεί να γίνει απευθείας χωρίς τον κίνδυνο να χαθεί κανένα μέρος της αρχικής πληροφορίας. Το πλεονέκτημα των δικτύων που



χρησιμοποιούν off line διαδικασίες εκπαίδευσης επικεντρώνεται κυρίως στη δυνατότητα να δίνουν καλύτερες λύσεις σε δύσκολα προβλήματα. ^{[36][37][38]}

4.3 Perceptron (Αισθητήρας)

Το μοντέλο του αισθητήρα (Perceptron) αποτελεί την απλούστερη μορφή νευρωνικού δικτύου. Απαρτίζεται νευρώνες τύπου McCulloch και Pitts με n κόμβους εισόδου και έναν κόμβο εξόδου. Χρησιμοποιεί ως συνάρτηση ενεργοποίησης τη βηματική συνάρτηση και μπορεί να διαχωρίσει πρότυπα δύο κλάσεων που είναι γραμμικά διαχωρίσιμες. Το μοντέλο αυτού του νευρωνικού δικτύου εκπαιδεύεται με επίβλεψη. Αρχικά ορίζονται οι στόχοι για την κάθε κλάση, θέτοντας την επιθυμητή τιμή στους κόμβους εξόδου. Εφόσον χρησιμοποιούμε τη βηματική συνάρτηση αυτοί είναι, 0 για την πρώτη κλάση και 1 για την δεύτερη. Κατά την εκπαίδευση εισάγονται τα πρότυπα με τη σειρά στο δίκτυο. Η διαδικασία αυτή αποκαλείται εποχή. Στη συνέχεια, η έξοδος συγκρίνεται με το στόχο και διορθώνονται οι συνάψεις. Η τροποποίηση των βαρών των συνάψεων γίνεται σύμφωνα με τον κανόνα δέλτα (Delta rule). Η εκπαίδευση ολοκληρώνεται όταν, κατά την επανάληψη, δεν μεταβάλλονται πλέον τα βάρη των συνάψεων.

Ο αισθητήρας αποτελείται από ένα δίκτυο κόμβων με δύο επίπεδα. Το πρώτο επίπεδο περιέχει τις εισόδους του δικτύου χωρίς να περιέχει νευρώνες. Αυτό σημαίνει πως δεν γίνεται κάποια επεξεργασία της πληροφορίας σε αυτό το επίπεδο. Το δεύτερο επίπεδο αποτελείται από νευρώνες και είναι το επίπεδο εξόδου του δικτύου. ^[38]



Κανόνας Δέλτα (delta rule)

Στη μηχανική μάθηση, ο κανόνας δέλτα αποτελεί ένα κανόνα μάθησης βασισμένο στη μέθοδο καθόδου βάσης της κλίσης (gradient descent), σύμφωνα με τον οποίο ανανεώνονται τα βάρη των εισόδων σε ένα TNN. Για έναν νευρώνα j με συνάρτηση ενεργοποίησης $g(x)$, ο κανόνας δέλτα για το συναπτικό βάρος w_{ji} δίνεται από την σχέση

$$\Delta w_{ji} = \alpha (t_j - y_j) g'(h_j) x_i,$$

όπου

α είναι μια σταθερά για τον συντελεστή εκμάθησης (learning rate)

$g(x)$ είναι η συνάρτηση ενεργοποίησης του νευρώνα

t_j είναι η επιθυμητή έξοδος

h_j είναι το άθροισμα βαρών των εισόδων των νευρώνων με

$$h_j = \sum x_i w_{ji}$$

y_j είναι η πραγματική έξοδος με

$$y_j = g(h_j)$$

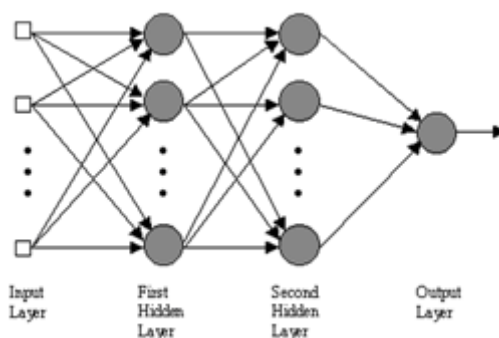
x_i είναι η $i_{\text{οστ}j}$ είσοδος

Ο κανόνας δέλτα αναπαρίσταται για νευρώνα με γραμμική συνάρτηση ενεργοποίησης πιο απλά ως εξής: ^[40]

$$\Delta w_{ji} = \alpha (t_j - y_j) x_i$$

4.4 Πολυεπίπεδοι αισθητήρες (Multilayer Perceptrons, MLPs)

Ενώ ο απλός αισθητήρας αποτελείται από ένα επίπεδο υπολογιστικών νευρώνων, οι πολυεπίπεδοι αισθητήρες αποτελούνται από περισσότερα από ένα επίπεδα υπολογιστικών νευρώνων. Τα επίπεδα αυτά που μεσολαβούν ανάμεσα στο επίπεδο εισόδου και το επίπεδο εξόδου ονομάζονται κρυφά επίπεδα (hidden layers). Το μοντέλο των πολυεπίπεδων αισθητήρων ανήκει στην κατηγορία των δικτύων εμπρόσθιας τροφοδότησης (feedforward networks). Η εκπαίδευση αυτών των νευρωνικών δικτύων γίνεται με επίβλεψη, καθώς απαιτούν να οριστεί επιθυμητή έξοδος για να εκπαιδευτούν.



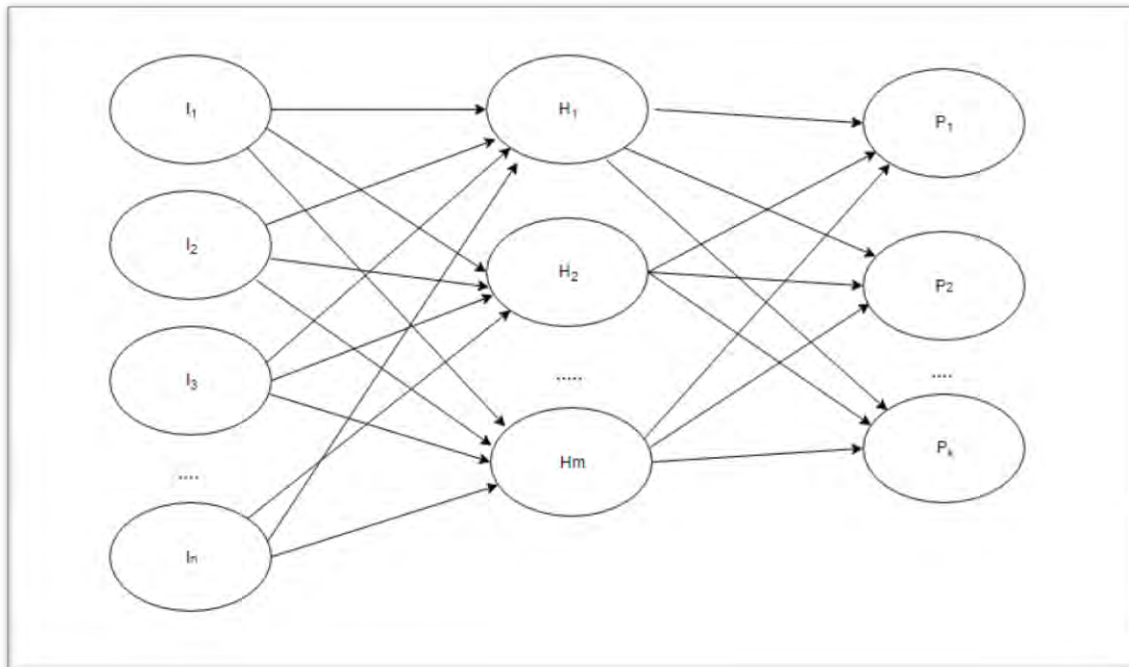
Εικόνα 4.4.1: Πολυεπίπεδος αισθητήρας με δύο κρυφά επίπεδα

Η ροή της πληροφορίας σε ένα τέτοιο μοντέλο γίνεται πάντοτε από τα αριστερά προς τα δεξιά. Το πρώτο επίπεδο αποτελείται από μη υπολογιστικούς νευρώνες που δέχονται την είσοδο και την προωθούν στο αμέσως επόμενο κρυφό επίπεδο. Η διαδικασία αυτή επαναλαμβάνεται μέχρι η πληροφορία να οδηγηθεί στην έξοδο, χωρίς να συναντήσει κάποιο βρόγχο ανάδρασης.^{[38] [41]}

4.5 Αλγόριθμος ανάστροφης διάδοσης (Back-propagation algorithm)

Ο αλγόριθμος Back-propagation αποτελεί την εξέλιξη του μοντέλου του απλού perceptron με την επέκταση του κανόνα δέλτα και παρέχει μια αποτελεσματική μέθοδο εκπαίδευσης των πολυεπίπεδων αισθητήρων.

Το μοντέλο νευρωνικού δικτύου αποτελείται από το επίπεδο εισόδου ή στοιβάδα εισόδου (input layer) που περιέχει n νευρώνες εισόδου, το κρυφό επίπεδο ή κρυφή στοιβάδα (hidden layer) που αποτελείται από m ενδιάμεσους νευρώνες και το επίπεδο εξόδου ή στοιβάδα εξόδου (output layer) που αποτελείται από k νευρώνες εξόδου.



Εικόνα 4.5.1: Τεχνητό Νευρωνικό δίκτυο με κρυφό επίπεδο



Μπορεί να υλοποιηθεί με συνεχείς συναρτήσεις ενεργοποίησης (σιγμοειδής (sigmoid), γραμμική (linear), υπερβολική εφαπτομένη (tan) κ.α.). Η πιο αποτελεσματική συνάρτηση ενεργοποίησης είναι η σιγμοειδής συνάρτηση (Sigmoid equation).

$$\sigma(x) = \frac{1}{1+e^{-x}}$$

Ο αλγόριθμος αυτό έχει τη δυνατότητα διαχωρισμού προτύπων 2 ή περισσότερων κλάσεων που είναι είτε γραμμικά είτε μη γραμμικά διαχωρίσιμες.

Πρόκειται για ένα μοντέλο νευρωνικού δικτύου που εκπαιδεύεται με επίβλεψη.

Απαραίτητο είναι ένα αντιπροσωπευτικό σετ προτύπων (training set) που θα δεχτεί σαν είσοδο το νευρωνικό δίκτυο κατά τη διάρκεια της εκπαίδευσης και ο ορισμός των στόχων.

Αρχικά θέτονται, λοιπόν, οι στόχοι του νευρωνικού δικτύου, δηλαδή οι επιθυμητές τιμές των κλάσεων ώστε να επιτευχθεί επιτυχημένη αναγνώριση. Οι τιμές αυτές εξαρτώνται από τη συνάρτηση ενεργοποίησης που έχει χρησιμοποιηθεί.

Επιθυμητό διάνυσμα εξόδου

$$p(x) = [p_1(x), p_2(x), \dots, p_k(x)]^T$$

Αρχικοποιούνται τα βάρη και τα κατώφλια των νευρώνων χρησιμοποιώντας μικρές θετικές τυχαίες τιμές.

Κατά την εκπαίδευση, εισάγεται το training set στο κρυφό επίπεδο από το οποίο η έξοδος χρησιμοποιείται σαν είσοδος για το επίπεδο εξόδου.

Training set – Διάνυσμα εκπαίδευσης

$$t(x) = [t_1(x), t_2(x), \dots, t_n(x)]^T$$

Υπολογίζονται όλα τα σήματα εξόδου των κόμβων (νευρώνων) προς τα εμπρός χρησιμοποιώντας τις τρέχουσες τιμές των συναπτικών βαρών.

$$y_i(x) = f_i(u_i(x))$$

$$u_i(x) = \sum_j w_{ij}(x)y_j(x)$$

όπου:

$y_j(x)$ είναι η έξοδος του j νευρώνα που γίνεται είσοδος του i νευρώνα

$w_{ij}(x)$ είναι το συναπτικό βάρος που συνδέει τον i με τον j νευρώνα

Για τους νευρώνες i πρώτου επιπέδου ισχύει



$$y_j(x) = t_j(x), \quad j = 1, 2 \dots n$$

Για τους νευρώνες j του επιπέδου εξόδου, το $y_j(x)$ είναι η j έξοδος του δικτύου.

Στη συνέχεια, ανανεώνονται τα βάρη, ξεκινώντας από το επίπεδο εξόδου και προχωρώντας αντίστροφα προς το επίπεδο εισόδου χρησιμοποιώντας τον κανόνα δέλτα για την ανανέωση των τιμών των συναπτικών βαρών

$$w_{ij}(x+1) = w_{ij}(x) + \gamma \delta_i(x) y_j(x)$$

Όπου

$\delta_i(x)$ είναι η τοπική κλίση και αν ο αναφερόμενος νευρώνας ανήκει στο επίπεδο εξόδου δίνεται από την ακόλουθη σχέση:

$$\delta_i(x) = [p_i(x) - y_i(x)] y_i(x) [1 - y_i(x)]$$

Ενώ αν ο νευρώνας ανήκει στο κρυφό επίπεδο δίνεται από την σχέση:

$$\delta_i(x) = y_i(x) [1 - y_i(x)] \sum_k \delta_k(x) w_{ki}(x)$$

Όπου

k : το σύνολο των νευρώνων εξόδου

$w_{ki}(x)$: το συναπτικό βάρος που συνδέει τον νευρώνα i ενός επιπέδου με τον νευρώνα k του επόμενου επιπέδου.

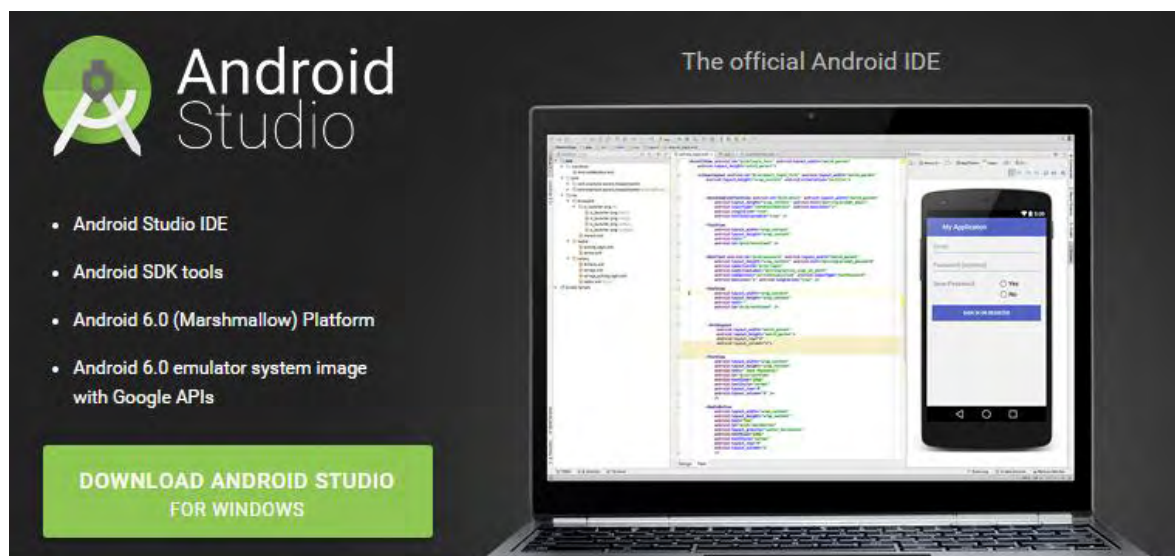
Η διαδικασία αυτή επαναλαμβάνεται έως ότου οι τιμές των εξόδων των νευρώνων του τελευταίου επιπέδου (νευρώνες εξόδου – output layer) πλησιάσουν στις επιθυμητές με ανοχή λάθους ϵ_0 .^[42]



5. Εργαλεία ανάπτυξης της εφαρμογής

5.1 Android Studio

Για την ανάπτυξη και τη σχεδίαση της εφαρμογής χρησιμοποιήθηκε το Android Studio 1.4. Είναι ένα ολοκληρωμένο προγραμματιστικό περιβάλλον (IDE) για προγραμματισμό εφαρμογών Android. Η πρώτη σταθερή έκδοση κυκλοφόρησε το Δεκέμβριο του 2014. Βασισμένο στο λογισμικό της JetBrains' IntelliJ IDEA, το Android Studio σχεδιάστηκε αποκλειστικά για Android προγραμματισμό. Είναι διαθέσιμο για Windows, Mac OS X και Linux και αντικατέστησε τα Eclipse Android Development Tools (ADT) ως κύριο IDE της Google για ανάπτυξη Android εφαρμογών. ^[10]



Εικόνα 5.1.1: Το Android Studio είναι διαθέσιμο για download στο <https://developer.android.com/sdk/index.html>

5.2 Android SDK

Το Android SDK είναι το software development kit της Google για όσους θέλουν να προγραμματίσουν για Android.

Συγκεκριμένα, είναι το επίσημο εργαλείο για τη δημιουργία εφαρμογών Android, τον προγραμματισμό custom ROM αλλά και για άλλες λειτουργίες όπως η μεταγλώττιση πυρήνα. Το Android SDK περιέχει ένα ολοκληρωμένο σύνολο προγραμματιστικών εργαλείων. Περιέχει τον debugger, βιβλιοθήκες, έναν προσομοιωτή βασισμένο στο QEMU, documentation, δείγματα κωδικών

κ.α. Υποστηρίζεται για συστήματα που τρέχουν

Linux, Mac OS X 10.5.8 και μεταγενέστερα

όπως και Windows XP και μεταγενέστερα. Από τον Μάρτη του 2015, το SDK δεν είναι διαθέσιμο για το ίδιο το Android παρόλο που ο προγραμματισμός υποστηρίζεται από ορισμένες μόνο εφαρμογές Android. Μέχρι το τέλος του 2014, το επίσημο περιβάλλον προγραμματισμού σε Android ήταν το Eclipse, το οποίο χρησιμοποιούσε το ADT plugin (Android Development Tools) ώστε να υποστηρίζει τον προγραμματισμό Android. Από το 2015, το Android Studio της Google που υποστηρίζεται από την IntelliJ είναι το επίσημο IDE.

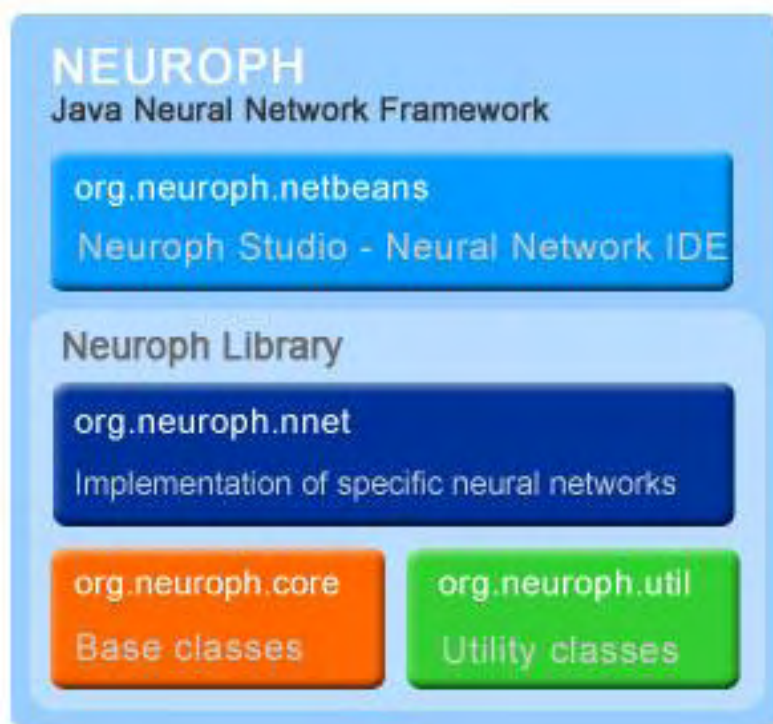
Το SDK αναβαθμίζεται συνεχώς παράλληλα με την αναβάθμιση της πλατφόρμας Android. Υποστηρίζει επίσης παλιότερες εκδόσεις της Android πλατφόρμας σε περίπτωση που οι προγραμματιστές επιθυμούν να φτιάξουν εφαρμογές που τρέχουν σε παλιότερες συσκευές. Τα εργαλεία προγραμματισμού είναι διαθέσιμα προς λήψη και ανεξάρτητα μεταξύ τους, οπότε μεταγενέστερες και προγενέστερες πλατφόρμες μπορούν να ενσωματωθούν και μετά την εγκατάσταση για δοκιμή της συμβατικότητας. ^{[11] [43]}



Εικόνα 5.2.1: Android SDK

5.3 Neuroph

Το Neuroph είναι ένα ελαφρύ Java framework για ανάπτυξη αρχιτεκτονικών νευρωνικών δικτύων. Περιέχει μια καλοσχεδιασμένη open-source βιβλιοθήκη σε Java



Εικόνα 5.3.1: Neuroph Framework

με έναν αριθμό βασικών κλάσεων που αντιστοιχούν σε βασικές έννοιες νευρωνικών δικτύων.

Επίσης περιλαμβάνει GUI editor, το Neuroph Studio που δίνει τη δυνατότητα να δημιουργηθούν γρήγορα συστατικά στοιχεία νευρωνικών δικτύων. Έχει κυκλοφορήσει ως open source project υπό την άδεια Apache 2.0 και είναι δωρεάν για χρήση.

Περιέχει βασικές κλάσεις για νευρωνικά δίκτυα που

αντιστοιχούν σε βασικά

συστατικά νευρωνικών δικτύων όπως είναι ο τεχνητός νευρώνας, το επίπεδο νευρώνων, η σύναψη, το βάρος, συνάρτηση ενεργοποίησης, συνάρτηση εισόδου, κανόνες μάθησης κ.α. Το Neuroph υποστηρίζει βασικές αρχιτεκτονικές νευρωνικών δικτύων όπως είναι οι πολυεπίπεδοι αισθητήρες, τα δίκτυα Kohonen και Hopfield. Οι κλάσεις αυτές μπορούν να γίνουν επέκταση και να χρησιμοποιηθούν για να φτιαχτούν πιο σύνθετες αρχιτεκτονικές και μέθοδοι μάθησης. Το Neuroph παρέχει επίσης και ενσωματωμένη υποστήριξη ειδικά για αναγνώριση εικόνας.^[12]

Το Neuroph υποστηρίζει πολλές αρχιτεκτονικές νευρωνικών δικτύων όπως οι παρακάτω:



- Adaline
- Perceptron
- Multi Layer Perceptron with Backpropagation, Momentum on Resilient Propagation
- Hopfield network
- Bidirectional Associative Memory
- Kohonen network
- Hebbian network
- Maxnet
- Competitive network
- Instar
- Outstar
- RBF network
- Neuro Fuzzy Reasoner

Το documentation του Neuroph είναι διαθέσιμο στην ακόλουθη διεύθυνση:
<http://neuroph.sourceforge.net/javadoc/index.html>

5.4 Neuroph Studio

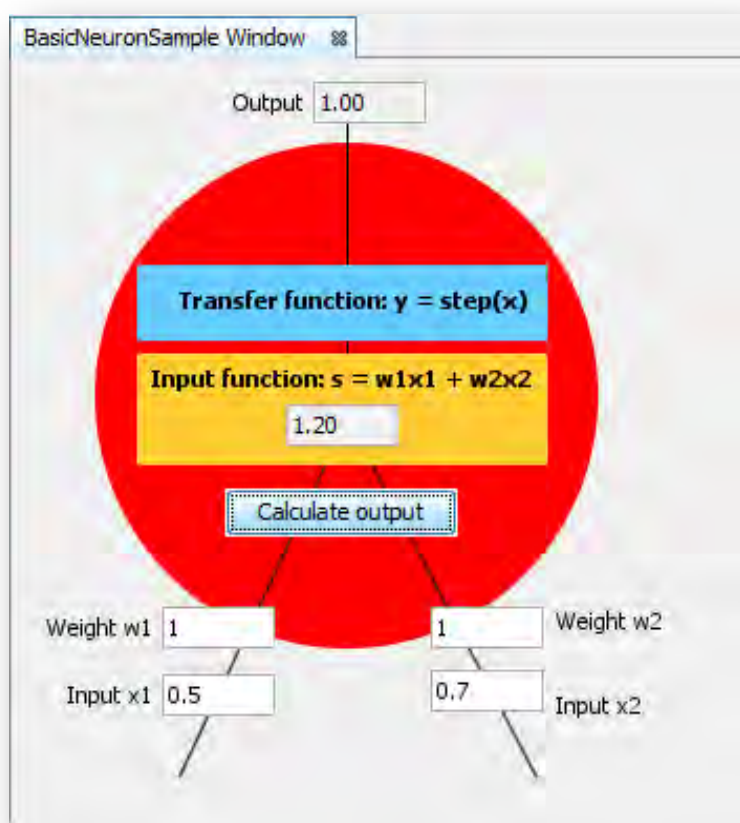
Το Neuroph Studio είναι ένα περιβάλλον ανάπτυξης νευρωνικών δικτύων σε Java. Στην ουσία αποτελεί το GUI editor του Neuroph. Απαιτείται να είναι εγκατεστημένη η έκδοση 1.7 της Java για να τρέξει. Για να εγκατασταθεί το NeurophStudio, απαιτείται η εκτέλεση της εφαρμογής (neurophstudio-windows.exe για windows ή neurophstudio-linux.sh για linux) και να ακολουθηθούν τα απλά βήματα που θα εμφανιστούν στο wizard.

Με το Neuroph Studio δίνεται η δυνατότητα δημιουργίας στοιχείων νευρωνικών δικτύων σε java.

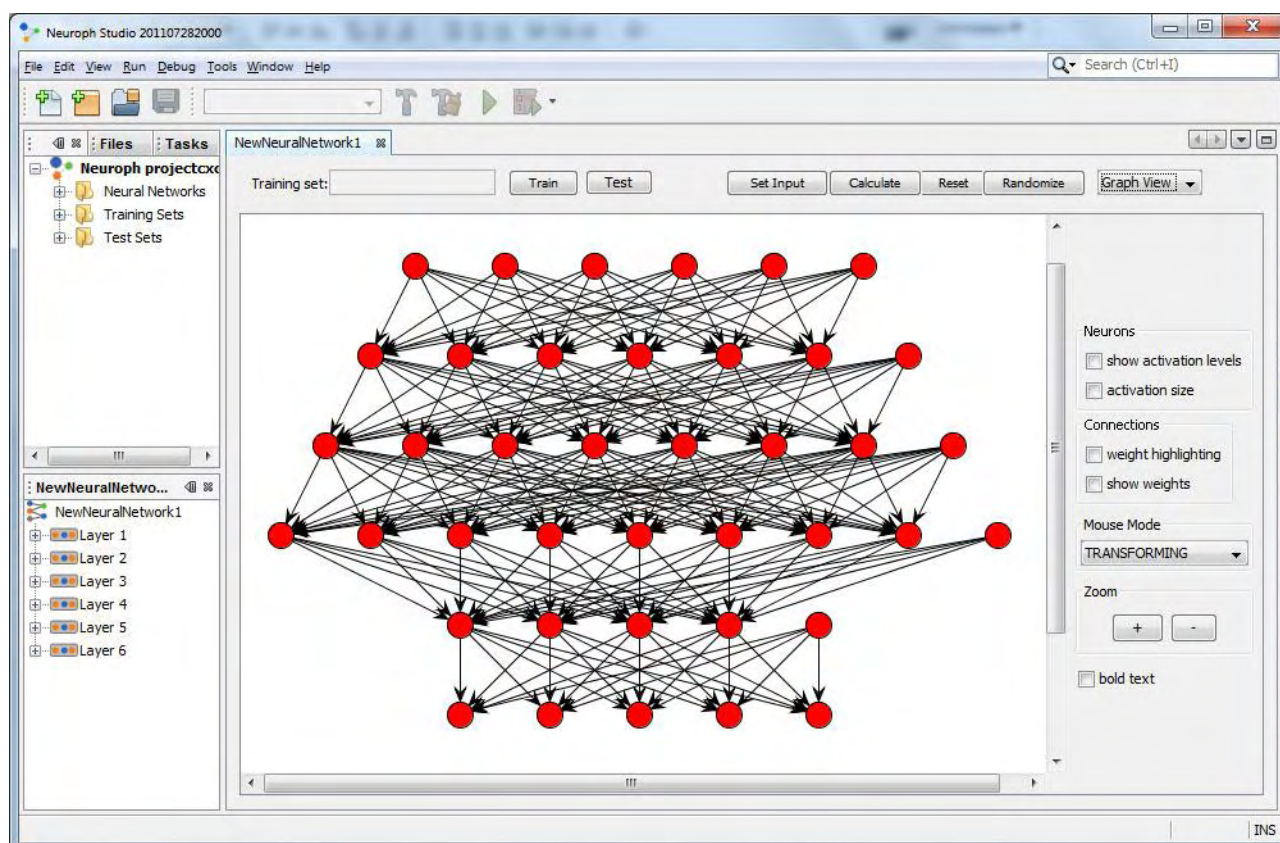
Εκτός από εύχρηστους οδηγούς και εργαλεία ανάπτυξης νευρωνικών δικτύων, το Neuroph Studio παρέχει και βασικά δομοστοιχεία Java του Netbeans IDE σε Java έτσι ώστε οι προγραμματιστές να μπορούν να δημιουργούν, να δοκιμάζουν και να εφαρμόζουν διάφορα συστατικά στοιχεία στο ίδιο περιβάλλον.

Το Neuroph Studio παρέχει ακόμη υποστήριξη για αναγνώριση εικόνας, αναγνώριση χαρακτήρων και χειρόγραφων γραμμάτων και για άλλες εξειδικευμένες λειτουργίες. Επίσης προσφέρει δείγματα κώδικα που παρουσιάζουν τις βασικές αρχές των νευρωνικών δικτύων.

Παρακάτω παρουσιάζεται ένα δείγμα των δυνατοτήτων του προγράμματος αυτού. ^[11]



Εικόνα 5.4.1: Παράδειγμα απλού νευρώνα



Εικόνα 5.4.2: Δείγμα αρχιτεκτονικής πολυεπίπεδων αισθητήρων



6. Υλοποίηση εφαρμογής

6.1 Αναγνώριση εικόνας με το Neuroph Studio

Το Neuroph Studio χρησιμοποιήθηκε για τη δημιουργία νευρωνικού δικτύου multi layer perceptron με σκοπό την αναγνώριση εικόνας. Το Neuroph περιέχει βιβλιοθήκη ειδικά για αναγνώριση εικόνας, ενώ το Neuroph Studio δίνει τη δυνατότητα δημιουργίας νευρωνικού δικτύου μέσω wizard.

Η διαδικασία αναγνώρισης βασίζεται στον τρόπο απεικόνισης μιας εικόνας ως δισδιάστατο πίνακα, όπου κάθε στοιχείο του περιέχει χρωματική πληροφορία για κάθε pixel. Κάθε χρώμα μπορεί να αναπαρασταθεί σαν συνδυασμός των τριών βασικών χρωμάτων με το σύστημα RGB (Red Green Blue). Για να αποθηκευτεί η συνολική πληροφορία μιας εικόνας σε ένα τέτοιο σύστημα, απαιτούνται 3 δισδιάστατοι πίνακες, ένας για κάθε βασικό χρώμα, όπου το κάθε στοιχείο αντιστοιχεί σε ένα pixel της εικόνας.

Int [][] redValues

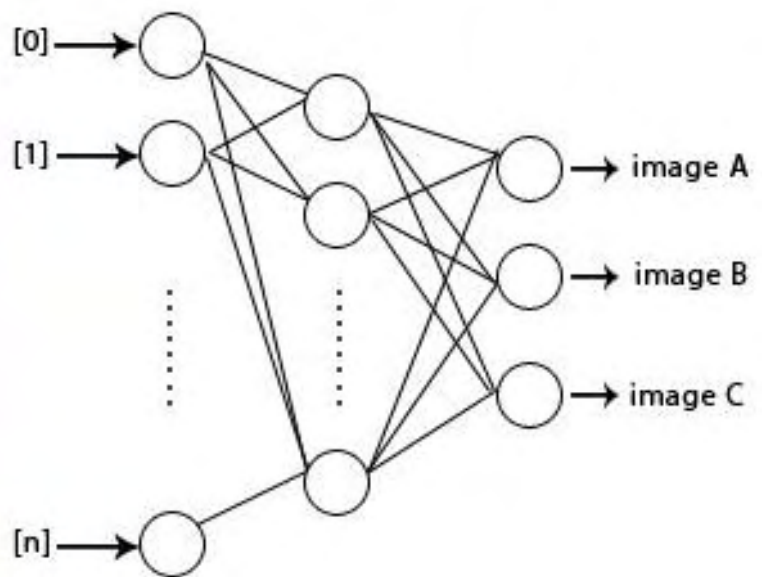
Int [][] greenValues

Int [][] blueValues,

Όπου οι διαστάσεις των πινάκων είναι [imageHeight][imageWidth]

Η πληροφορία αυτή υπάρχει και συγκεντρωμένη σε ένα μονοδιάστατο πίνακα τον flattenedRgbValues[imageHeight*imageWidth*3]. Ο πίνακας αυτός χρησιμοποιείται σαν είσοδος στο νευρωνικό δίκτυο ώστε να εκπαιδευτεί στην αναγνώριση ή την ομαδοποίηση των εικόνων.

Multi layer perceptrons είναι ο καταλληλότερος τύπος νευρωνικών δικτύων κατάλληλος για αυτή τη διαδικασία.

flattenedRgbValues[n] $n = \text{height} * \text{width} * 3$ 

Εικόνα 6.1.1: Μοντέλο αναπαράστασης εικόνας σε μονοδιάστατο πίνακα και νευρωνικού δικτύου

Σύμφωνα με την παραπάνω εικόνα ο κάθε νευρώνας εισόδου αντιστοιχεί με ένα χρωματικό στοιχείο συστήματος RGB του κάθε pixel της εικόνας σε συγκεκριμένη θέση του πίνακα αναπαράστασης. Αντιστοίχως, ο κάθε νευρώνας εξόδου αντιστοιχεί σε μία εικόνα ή μια κλάση εικόνων. Οπότε, μια έξοδο [1,0,0] του νευρωνικού δικτύου σημαίνει πως η εικόνα που δόθηκε σαν είσοδος στο νευρωνικό δίκτυο αναγνωρίζεται σαν εικόνα A (image A). Το νευρωνικό δίκτυο εκπαιδεύτηκε με τον αλγόριθμο Backpropagation. ^[12]



6.1.1 Δημιουργία νευρωνικού δικτύου

Όπως προαναφέρθηκε το Neuroph Studio παρέχει περιβάλλον δημιουργίας και εκπαίδευσης νευρωνικών δικτύων που μπορούν να χρησιμοποιηθούν σαν συστατικά στοιχεία σε κώδικα Java. Επιπλέον, δίνει τη δυνατότητα δημιουργίας νευρωνικού δικτύου εξειδικευμένο στην αναγνώριση εικόνας σε 5 στάδια.

1. Δημιουργία Neuroph project
2. Δημιουργία νευρωνικού δικτύου αναγνώρισης εικόνας
3. Εκπαίδευση νευρωνικού δικτύου
4. Δοκιμή νευρωνικού δικτύου
5. Αποθήκευση και ενσωμάτωση του δικτύου στον αλγόριθμο της αναγνώρισης εικόνας.

Για τη δημιουργία του training set χρησιμοποιήθηκαν συνολικά 20 εικόνες-δείγματα: 2 για κάθε χαρτονόμισμα ευρώ που κυκλοφορεί (παλιά και καινούρια).



Σύνολο δειγμάτων



Πίνακας 6.1.1: Εικόνες δείγματα του training set

Οι εικόνες του training set είναι σε διαστάσεις 120x64.



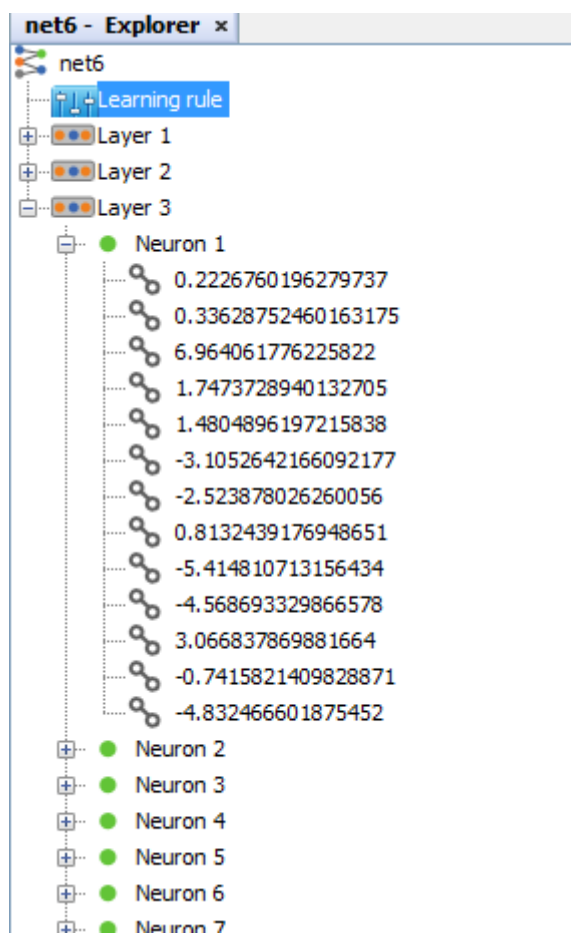
Οι διαστάσεις επιλέχτηκαν έτσι ώστε να έχουμε τη μέγιστη δυνατή αναγνώριση, χωρίς να υπερβαίνουμε το μέγεθος του νευρωνικού δικτύου που μπορεί να υποστηρίξει μια Android συσκευή.

Δίνεται επίσης η δυνατότητα να επιλεχθούν εικόνες προς αποφυγή αναγνώρισης. Συνήθως δίνονται μονόχρωμες εικόνες ώστε να βοηθηθεί η διαδικασία της αναγνώρισης.

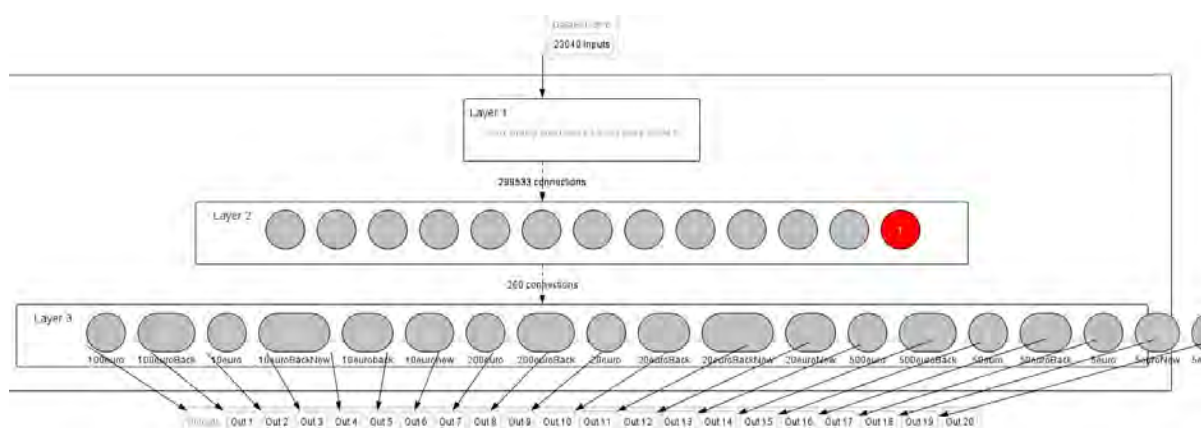
Καθορίζεται το training set καθώς και το sampling resolution. Η τιμή που θα δοθεί είναι η διάσταση στην οποία θα μετατραπούν οι εικόνες για την εκπαίδευση του νευρωνικού δικτύου. Η διάσταση που επιλέχθηκε για τον σκοπό της εργασίας αυτής μετά από δοκιμές είναι 120x64.

Πριν την ολοκλήρωση της διαδικασίας δημιουργίας του νευρωνικού δικτύου και του σετ εκπαίδευσης, ζητείται να επιλεγεί η συνάρτηση μεταφοράς. Οι επιλογές που παρουσιάζονται είναι η σιγμοειδής και η υπερβολική εφαπτομένη. Για τον σκοπό της εργασίας αυτής καλύτερα αποτελέσματα παρουσίασε η σιγμοειδής συνάρτηση.

Τέλος, σημαντικό βήμα είναι η επιλογή του αριθμού των κρυφών επιπέδων αλλά και ο αριθμός των νευρώνων που θα περιέχει το κάθε επίπεδο. Μετά από αρκετές δοκιμές, η επιλογή ενός κρυφού επιπέδου με 12 νευρώνες παρουσίασε τον καλύτερο χρόνο εκπαίδευσης όμως και ικανοποιητικά αποτελέσματα αναγνώρισης.

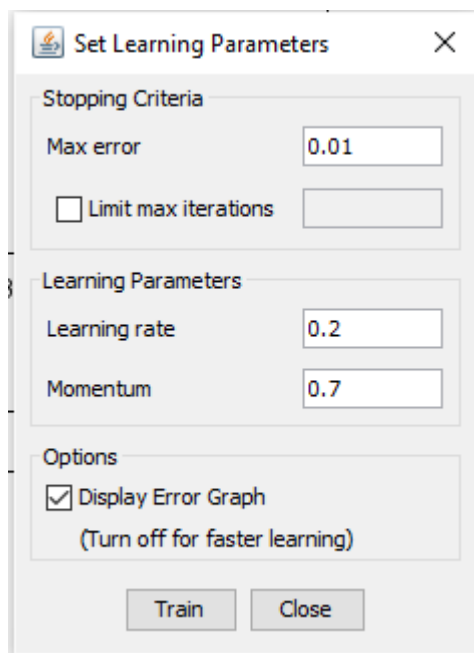


Εικόνα 6.1.2: Πλαίσιο εξερεύνησης των μαθηματικών τιμών των νευρώνων στο NeurophStudio



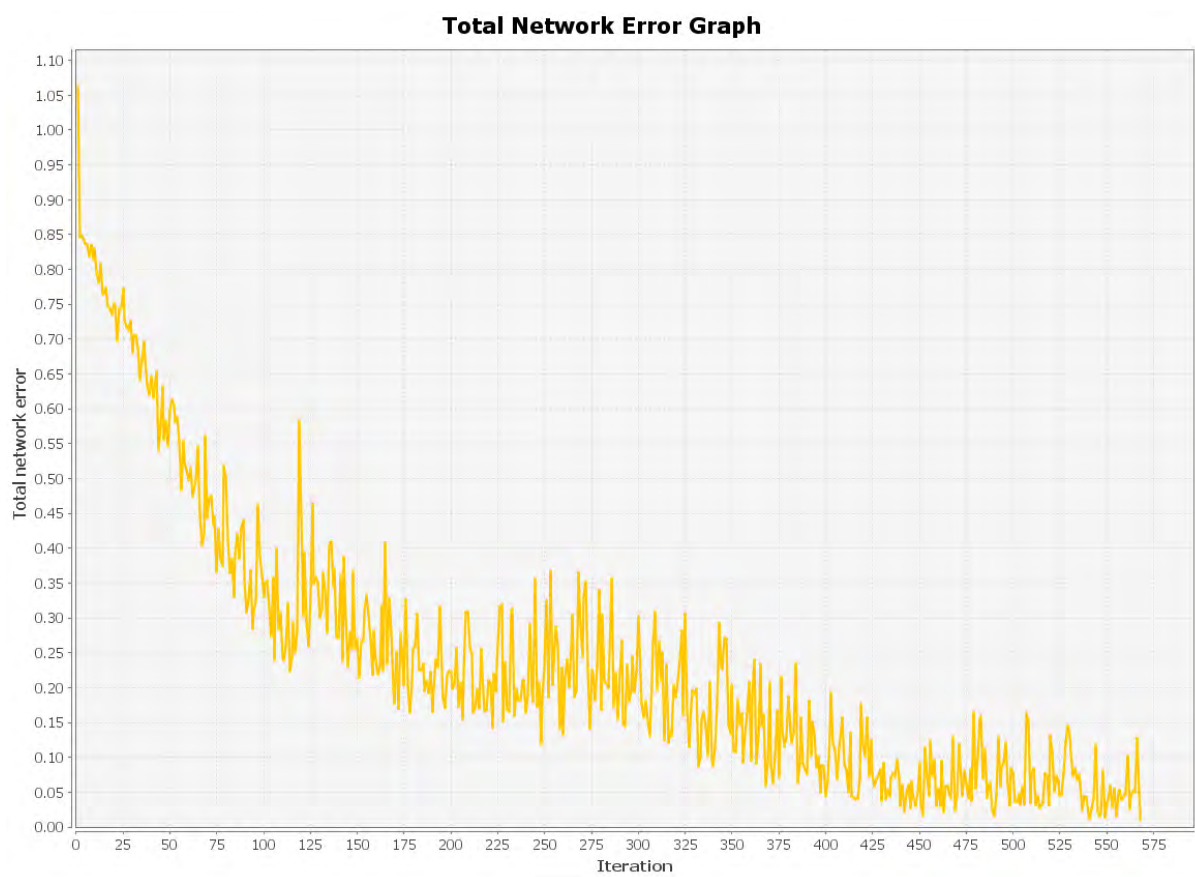
Εικόνα 6.1.3: Αρχιτεκτονική του νευρωνικού δικτύου

Αφού ολοκληρώθηκε η διαδικασία δημιουργίας νευρωνικού δικτύου, αρχίζει η διαδικασία της εκπαίδευσης του με βάση το training set που ορίστηκε με τον αλγόριθμο back-propagation. Απαιτείται η αρχικοποίηση των παραμέτρων όπως είναι το μέγιστο επιτρεπόμενο σφάλμα σε 0.01. Καθοριστικές για την αποτελεσματικότητα του αλγορίθμου είναι δυο συντελεστές: learning rate και momentum. Ο ρυθμός εκμάθησης του δικτύου (learning rate) ορίστηκε ως 0.2 ενώ το momentum, ο ρυθμός μεταβολής των συναπτικών βαρών, 0.7.



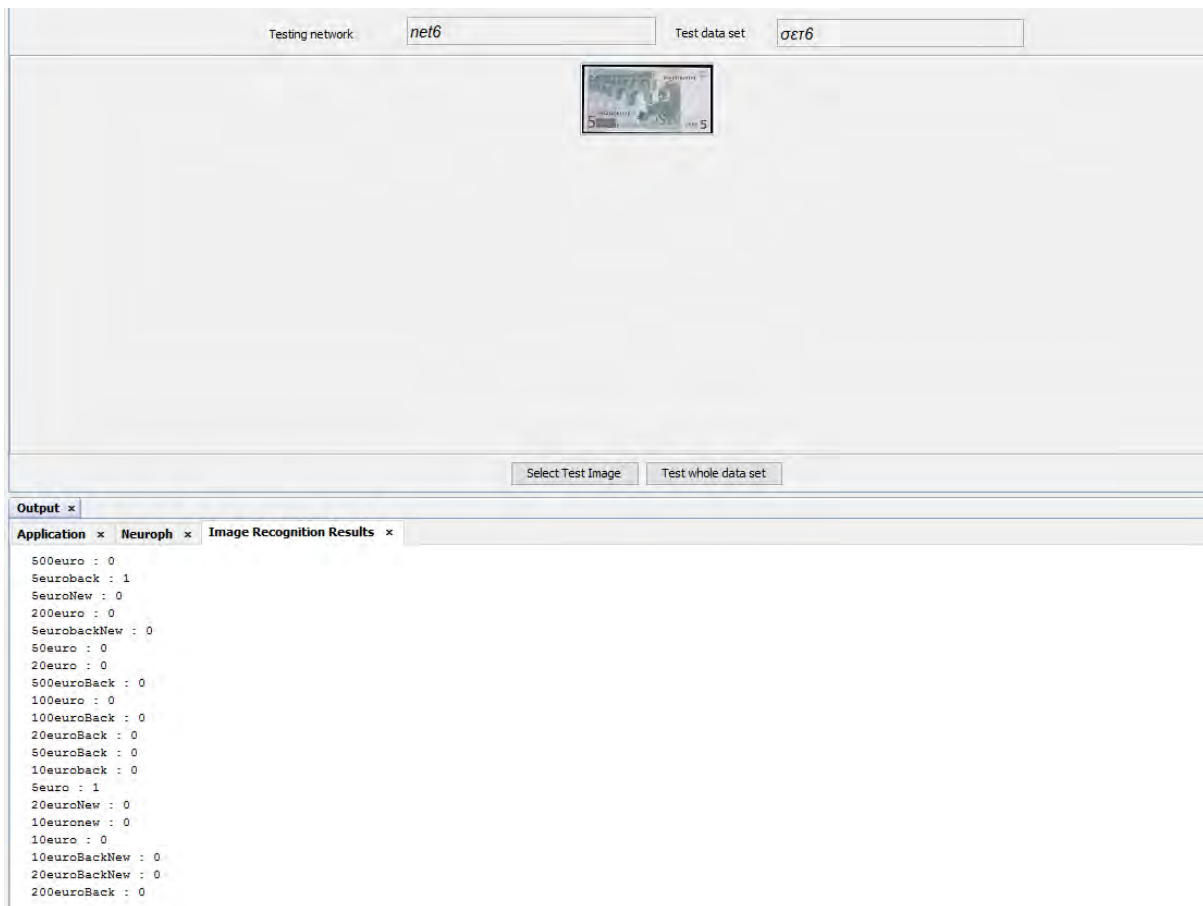
Εικόνα 6.1.4: Πάνελ αρχικοποίησης των συντελεστών του αλγορίθμου

Κατά τη διάρκεια της εκπαίδευσης του νευρωνικού δικτύου εμφανίζεται το διάγραμμα λάθος ως προς τον αριθμό επαναλήψεων(epochs)



Πίνακας 6.1.5: Αναπαράσταση σφάλματος νευρωνικού δικτύου ως προς τις επαναλήψεις κατά τη διάρκεια της εκπαίδευσης

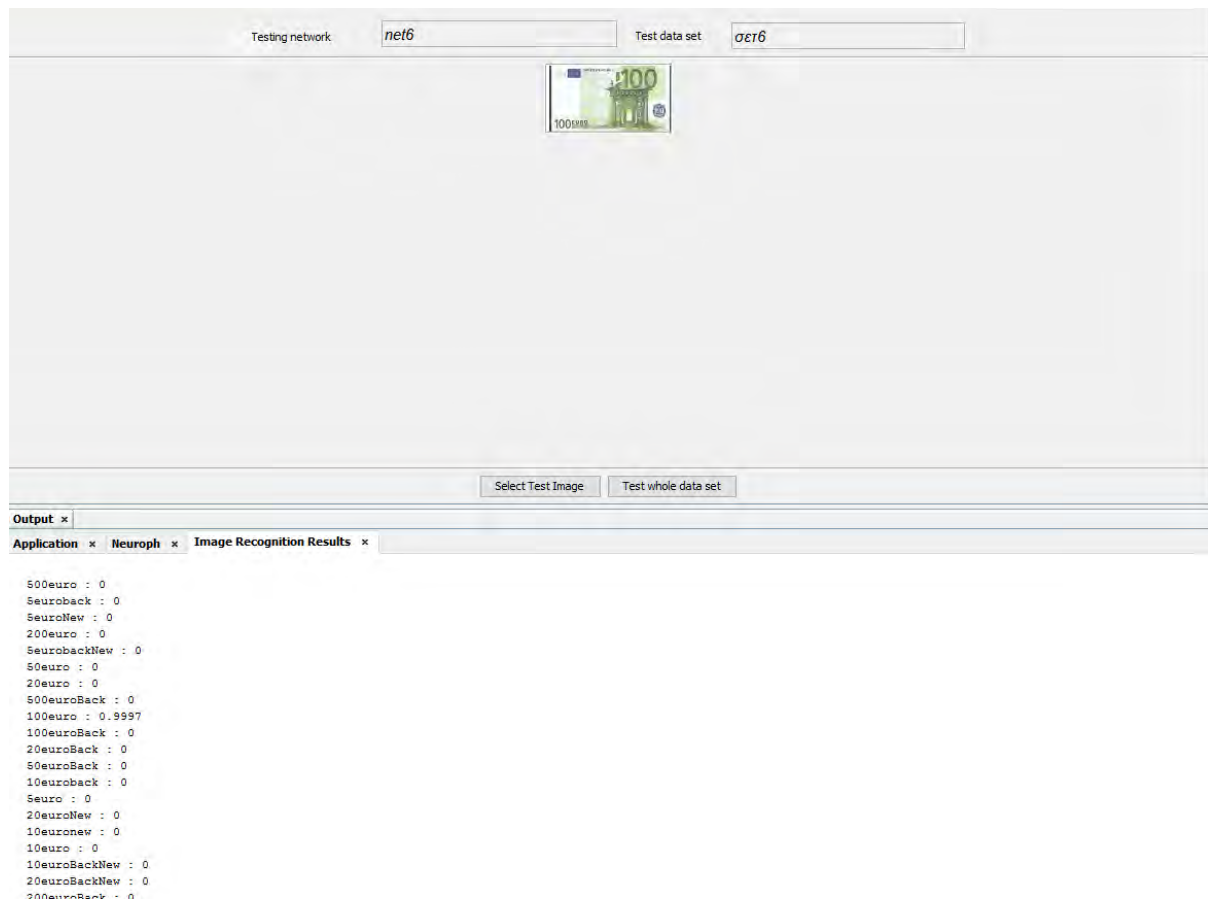
Αφού ολοκληρώθηκε η εκπαίδευση του νευρωνικού δικτύου στις 569 επαναλήψεις, δοκιμάζεται η αποτελεσματικότητα της αναγνώρισης που πετυχαίνει το νευρωνικό δίκτυο. Στην παρακάτω εικόνα, εισάγεται ως είσοδος στο δίκτυο, μια εικόνα από το training set την οποία αναγνωρίζει χωρίς κανένα σφάλμα. Εισάγεται μια εικόνα ενός χαρτονομίσματος των 5 ευρώ στο νευρωνικό δίκτυο. Η έξοδος που παράγει για την εικόνα αυτή είναι η τιμή 1 για τον νευρώνα εξόδου που αντιστοιχεί στην πίσω όψη του παλιού χαρτονομίσματος των 5 ευρώ, ενώ όλοι οι υπόλοιποι νευρώνες δίνουν ως έξοδο 0.



Εικόνα 6.1.5: Αποτελέσματα αναγνώρισης χαρτονομίσματος από το dataset

Το νευρωνικό αυτό δίκτυο παρουσίασε 100% πετυχημένη αναγνώριση στο training set.

Στη συνέχεια, δοκιμάστηκαν και εικόνες εκτός training set, στις οποίες είχε εξίσου εντυπωσιακά αποτελέσματα όπως φαίνεται στην παρακάτω εικόνα, στην οποία το νευρωνικό δέχεται σαν είσοδο εικόνα χαρτονομίσματος 100 ευρώ και ο νευρώνας εξόδου για το αντίστοιχο χαρτονόμισμα παίρνει την τιμή 0.9997.



Εικόνα 6.1.6: Αποτελέσματα αναγνώρισης χαρτονομίσματος εκτός του dataset



6.2 Λειτουργία εφαρμογής

Η λειτουργία της εφαρμογής αυτής σχεδιάστηκε να είναι απλή και εύχρηστη για άτομα που έχουν προβλήματα στην όραση.

Με την εκκίνηση της εφαρμογής εμφανίζεται η αρχική οθόνη, στην οποία περιέχεται μήνυμα καλωσορίσματος και ένα ευδιάκριτο κόκκινο κουμπί. Με το πάτημα του κουμπιού αυτού, γίνεται εκκίνηση της λειτουργίας της κάμερας του τηλεφώνου.

Αφού τραβήξει ο χρήστης τη φωτογραφία και επιλέξει να την αποθηκεύσει ακούγεται ηχητικό μήνυμα που ζητά από τον χρήστη να περιμένει όσο εκτελείται η διαδικασία της αναγνώρισης.

Μόλις ολοκληρωθεί η διαδικασία της αναγνώρισης στο background, ακούγεται ηχητικό μήνυμα που ενημερώνει το χρήστη για το αποτέλεσμα της αναγνώρισης. Σε περίπτωση που η φωτογραφία δεν είναι ευκρινής ή το αντικείμενο δεν αναγνωρίζεται εμφανίζεται σχετικό μήνυμα που ζητά από τον χρήστη να δοκιμάσει ξανά.

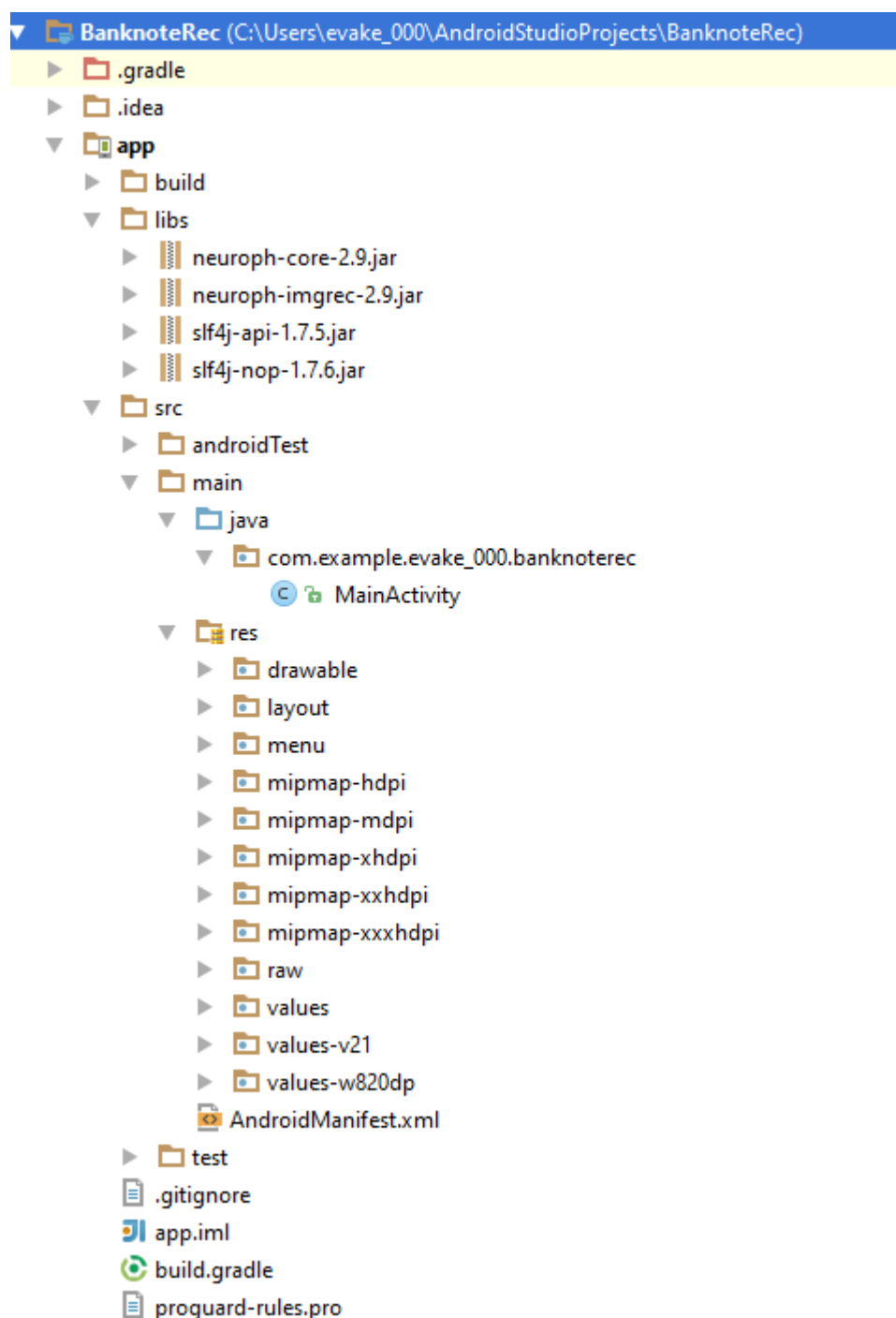


Εικόνα 6.2.1: Στιγμιότυπο εφαρμογής



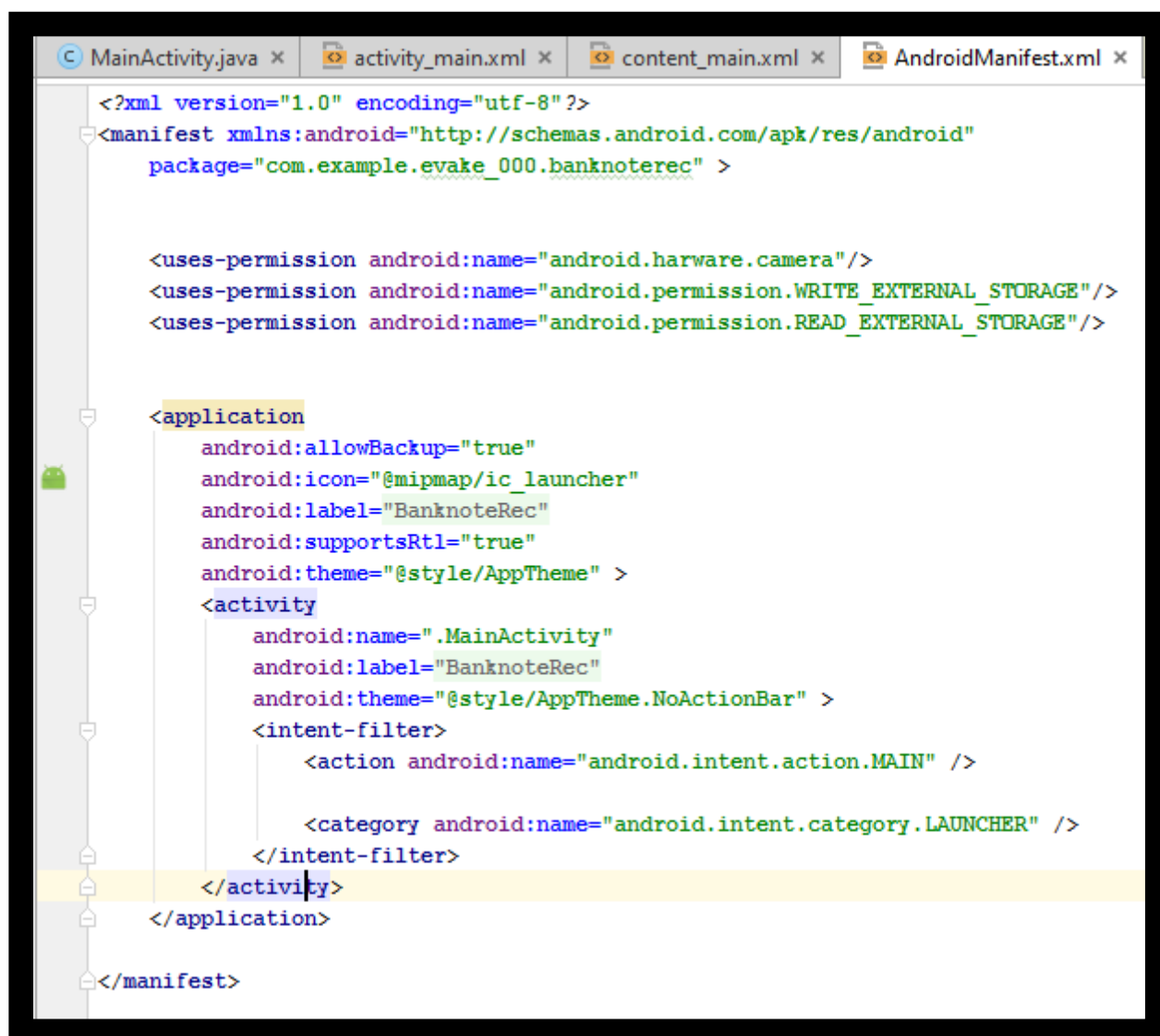
6.3 Δομή της εφαρμογής

Εξετάζοντας τη δομή της εφαρμογής BacknoteRec, εμφανίζονται οι βιβλιοθήκες που έχουν χρησιμοποιηθεί από το Neuroph API στον φάκελο /app/libs. Οι βιβλιοθήκες που είναι απαραίτητες για την υλοποίηση της αναγνώρισης εικόνας είναι οι neuroph-core-2.9.jar και neuroph-imgrec-2.9.jar, οι οποίες περιέχουν βασικές κλάσεις για τη δημιουργία, το στήσιμο και την εκπαίδευση του νευρωνικού δικτύου καθώς και τη διαδικασία της αναγνώρισης και της επιστροφής αποτελεσμάτων. Επίσης χρησιμοποιήθηκε το slf4j για logging.



Εικόνα 6.3.1: Φάκελοι και υποφάκελοι εφαρμογής

Στον κατάλογο /src/res βρίσκονται όλα τα απαραίτητα αρχεία για να τρέξει η εφαρμογή, όπως εικόνες στο /src/res/drawable (button.png, button_corners.xml, wait.png). Στον φάκελο αυτό βρίσκεται και το AndroidManifest.xml που περιέχει όλες τις άδειες που απαιτούνται για το τρέξιμο της εφαρμογής.



Εικόνα 6.3.2: AndroidManifest.xml

Στον φάκελο /src/res/raw υπάρχει το αρχείο του εκπαιδευμένου νευρωνικού δικτύου και τα αρχεία .mp3 για την ενημέρωση του χρήστη για το αποτέλεσμα της αναγνώρισης.



Εικόνα 6.3.3: Layout εφαρμογής (content_main.xml)



Η βασική λειτουργία της εφαρμογής περιγράφεται από την MainActivity.java.

Περιέχει τις εξής μεθόδους:

OnCreate()

Μέθοδος η οποία καλείται με την εκκίνηση της εφαρμογής. Εμφανίζει το layout , συνδέει το κουμπί με το listener και προσδιορίζει μια μεταβλητή του συστήματος με την System.setProperty() που είναι αναγκαία για την λειτουργία του API του Neuroph.

OnClickListener()

Μέθοδος η οποία ενεργοποιείται με το πάτημα του button_camera

onCreateOptionsMenu()

Η μέθοδος αυτή περιέχει μια κλήση της getMenuInflater().inflate() για την εμφάνιση xml μενού.

takePhoto()

Η μέθοδος αυτή είναι υπεύθυνη για την εκκίνηση της κάμερας του κινητού. Επίσης περιέχει κλήσεις για την αποθήκευση της φωτογραφίας στον εξωτερικό αποθηκευτικό χώρο φωτογραφιών του κινητού και την αποθήκευση του imageUri για μεταγενέστερη χρήση.

onActivityResult()

Η μέθοδος αυτή, ανάλογα το αποτέλεσμα της λειτουργίας της κάμερας, εμφανίζει την φωτογραφία καθώς και την πραγματική της τοποθεσία στον αποθηκευτικό χώρο του κινητού.

Παράλληλα κάνει δημιουργία και εκκίνηση του νήματος loadDataRunnable με stack size 64000.

Αφου ολοκληρωθεί το loadDataRunnable που υπολογίζει το αποτέλεσμα της αναγνώρισης, γίνεται αναπαραγωγή ηχητικού μηνύματος που ανακοινώνει το χαρτονόμισμα που αναγνωρίστηκε, εφόσον η πιθανότητα της αναγνώρισης είναι μεγαλύτερη από 0.32.



Διαφορετικά παρουσιάζεται μήνυμα αδυναμίας αναγνώρισης του αντικειμένου της φωτογραφίας και ζητείται από τον χρήστη να δοκιμάσει ξανά.

loadDataRunnable()

Το thread αυτό είναι υπεύθυνο για τη λειτουργία της αναγνώρισης εικόνας. Δημιουργήθηκε με stack 64000 ώστε να μπορεί να φορτώσει το εκπαιδευμένο νευρωνικό δίκτυο. Να σημειωθεί εδώ, πως είναι δυνατή η δημιουργία και η εκπαίδευση νευρωνικού δικτύου μέσω του Neuroph API. Μόλις αυτό φορτωθεί, τυπώνεται σχετικό μήνυμα για τον προγραμματιστή με τον αριθμό των επιπέδων (3). Στη συνέχεια φορτώνεται το imageRecognition plugin από την ImageRecognitionPlugin.class. Κατόπιν, γίνεται πρόσβαση στο file path της φωτογραφίας που τράβηξε ο χρήστης, μέσω του Uri. Δημιουργείται ένα mutable bitmap που περιέχει την φωτογραφία, ώστε να μπορεί να γίνει επεξεργασία, με διαστάσεις σύμφωνες με των εικόνων του training set.

ImageFactory

Γίνεται δημιουργία νέου αρχείου εικόνας μέσω της κλάσης ImageFactory. Η κλάση αυτή δημιουργήθηκε για να παράγει αντικείμενα εικόνας τύπου Image ώστε να υποστηρίζονται και από την Android πλατφόρμα αλλά και από την J2SE. Να σημειωθεί εδώ πως ο λόγος που γίνεται αυτό, είναι επειδή η κλάση bufferedImage δεν μπορεί να χρησιμοποιηθεί στο Android. Το Android SDK δεν περιέχει τη βιβλιοθήκη javax.imageio αλλά υποστηρίζει την κλάση Bitmap.

Η κλάση ImageFactory περιέχει 3 overwritten κλάσεις getImage όπου δίνεται η δυνατότητα να επιστραφεί αρχείο εικόνας Image παίρνοντας σαν είσοδο είτε τύπο αρχείου File, String filepath είτε Uri.

```
private Runnable loadDataRunnable = new Runnable() {
    NeuralNetwork nnet = null;
    public void run() {
        // create image file
        Image banknoteImage;

        // load neural network file
        InputStream is = getResources().openRawResource(R.raw.bas_nnet);
        // load neural network
        nnet = NeuralNetwork.load(is);

        System.out.println("##### Network is loaded. Number of layers:" +
            nnet.getLayersCount() + " #####");

        // get the image recognition plugin from neural network
        imageRecognition = (ImageRecognitionPlugin) nnet.getPlugin(ImageRecognitionPlugin.class);
        // load image
        Uri selectedImage = imageUri;
        // store file path of selectedImage
        String filePath = selectedImage.getPath();
        System.out.println("##### Image file path: " + filePath + " #####");

        //Get picture's directory
        File dir=Environment.getExternalStoragePublicDirectory(Environment.DIRECTORY_PICTURES);

        //Load bitmap from filepath
        Bitmap b=BitmapFactory.decodeFile(filePath);
        // Resize bitmap in accordance with training set 120x64
        Bitmap out = Bitmap.createScaledBitmap(b, 120, 64, false);
        //Make mutable copy of bitmap
        Bitmap mutableBitmap=out.copy(Bitmap.Config.ARGB_8888, true);

        // create png file
        File newfile = new File(dir, "resize.png");

        // save image after resizing
        FileOutputStream fOut;
        try {
            fOut = new FileOutputStream(newfile);
            mutableBitmap.compress(Bitmap.CompressFormat.PNG, 100, fOut);
            fOut.flush();
            fOut.close();
            b.recycle();
            out.recycle();
            mutableBitmap.recycle();
        } catch (Exception e) {}

        // load banknoteImage
        banknoteImage = ImageFactory.getImage(newfile);
        //makes all the necessary calculations so that Neural network takes banknoteImage as input
        imageRecognition.setInput(banknoteImage);
    }
}
```

[...]

Εικόνα 6.3.4: Κώδικας loadDataRunnable()



setInput()

Μετά την δημιουργία της εικόνας τύπου Image, γίνεται κλήση της setInput η οποία δέχεται σαν είσοδο την εικόνα αυτή. Με την κλήση της συνάρτησης, αρχικά γίνεται downsample της εικόνας ώστε να συμβαδίζει με το resolution που έγινε η εκπαίδευση του νευρωνικού δικτύου. Στη συνέχεια, καλείται η συνάρτηση getFlattenedRgbValues() που συγκεντρώνει όλα τα στοιχεία της εικόνας σε πίνακα μιας διάστασης.

processInput()

Στη συνέχεια καλείται η processInput() η οποία είναι υπεύθυνη για τον υπολογισμό των τιμών των νευρώνων.

getOutput()

Τέλος καλείται η getOutput(), οποία κατασκευάζει και επιστρέφει ένα HashMap<String,Double> όπου περιέχονται οι τιμές των νευρώνων εξόδου καθώς και οι ετικέτες με το όνομα του χαρτονομίσματος που αντιστοιχεί ο κάθε νευρώνας.

Το thread ολοκληρώνεται αφού επιστραφούν τα αποτελέσματα και επεξεργαστούν ώστε να υπολογιστεί το ζευγάρι <τιμής, ετικέτας> του HashMap που η τιμή του νευρώνα εξόδου θα είναι η μεγαλύτερη από αυτές που περιέχει. Το αποτέλεσμα της διαδικασίας αυτής αποθηκεύεται στη μεταβλητή result.



6.4 Απαιτήσεις εφαρμογής

Για την εγκατάσταση της εφαρμογής απαιτείται έκδοση λογισμικού Android API 17 ή μεταγενέστερο. Δεν υπάρχει περιορισμός στην ανάλυση της οθόνης.



7. Συμπεράσματα

7.1 Δοκιμές εφαρμογής

Για να διαπιστωθεί η σωστή λειτουργία της εφαρμογής πραγματοποιήθηκαν δοκιμές σε τρεις συσκευές Android. Δεν χρησιμοποιήθηκε κάποια εικονική συσκευή Android λόγω της ανάγκης συμμετοχής της κάμερας του κινητού.

Μετά την ολοκλήρωση της εφαρμογής, πραγματοποιήθηκαν δοκιμές στις παρακάτω συσκευές για τον εντοπισμό πιθανών σφαλμάτων. Αφού έγιναν κάποιες προσαρμογές στα γραφικά στοιχεία της εφαρμογής και κάποιες τροποποιήσεις στο νευρωνικό δίκτυο, προέκυψαν τα επιθυμητά αποτελέσματα.

Αριθμός Μοντέλου	Lenovo S90-u	Redmi Note	Samsung Galaxy Tab 2.10.1 P5100
Έκδοση Android	4.4.4	4.4.2	4.0.3
Επεξεργαστής	Quad-core 1.2GHz	Quad-core 1.6GHz	Dual-core 1GHz
RAM	2GB	2GB	1GB
Μνήμη συσκευής	16GB	8GB	16GB
Μέγεθος οθόνης	5" HD (1280 x 720)	5.5" HD (1280 x 720)	10.1" (800 x 1280)
Κάμερα	13MP	13MP f/2.2, 28mm	3.15MP

Πίνακας 7.1: Χαρακτηριστικά συσκευών που χρησιμοποιήθηκαν στις δοκιμές



7.2 Εξέλιξη της εφαρμογής

Αν και τα αποτελέσματα της εφαρμογής είναι ικανοποιητικά, υπάρχει δυνατότητα βελτίωσης. Αρχικά, θα ήταν σημαντικό αλλά και πολύ ενδιαφέρον να επικεντρωθεί κανείς στην βελτίωση της ταχύτητας της εφαρμογής ώστε να προκύπτουν γρηγορότερα τα αποτελέσματα της αναγνώρισης.

Επιπλέον, θα μπορούσε να επεκταθεί η εφαρμογή αυτή για αναγνώριση χαρτονομισμάτων και άλλων συναλλαγμάτων εκτός του Ευρώ. Δεδομένου ότι η δυνατότητα που παρέχει αυτή η εργασία θα διευκολύνει πολύ την καθημερινότητα των ανθρώπων που έχουν προβλήματα στην όραση, με την επέκταση αυτή, θα μπορούσε να απευθυνθεί σε μεγαλύτερο κοινό.

Τέλος, όσο αφορά το γραφικό περιβάλλον της εφαρμογής, θα μπορούσαν να γίνουν κάποιες τροποποιήσεις ώστε να γίνει πιο ευχάριστη στο χρήστη.



Ευρετήριο Εικόνων

Εικόνα 2.1.1: Κατοχή έξυπνων τηλεφώνων

Εικόνα 2.1.2: Χρήστες έξυπνων τηλεφώνων και ηλεκτρονικών υπολογιστών σε παγκόσμιο επίπεδο

Εικόνα 2.2.1: Λογότυπο Android

Εικόνα 2.2.1.1: Λογότυπο Android market

Εικόνα 2.2.2.1: Το λειτουργικό Android αποτελεί το πιο διαδεδομένο λειτουργικό για κινητά.

Εικόνα 2.2.2.2: Λογότυπο Android Cupcake

Εικόνα 2.2.2.3: Λογότυπο Android Donut

Εικόνα 2.2.2.4: Λογότυπο Android Eclair

Εικόνα 2.2.2.5: Λογότυπο Android Froyo

Εικόνα 2.2.2.6: Λογότυπο Android Gingerbread

Εικόνα 2.2.2.7: Λογότυπο Android Honeycomb

Εικόνα 2.2.2.8: Λογότυπο Android Ice Cream Sandwich

Εικόνα 2.2.2.9: Λογότυπο Android Jelly Bean

Εικόνα 2.2.2.10: Λογότυπο Android KitKat

Εικόνα 2.2.2.11: Λογότυπο Android Lollipop

Εικόνα 2.2.3.1: Εφαρμογές Android

Εικόνα 2.2.4.1: Αρχιτεκτονική Android

Εικόνα 3.1.1: Παράδειγμα αρχιτεκτονικής SOM

Εικόνα 4.1: Νευρωνικά δίκτυα, προσημείωση της λειτουργίας του ανθρώπινου εγκεφάλου

Εικόνα 4.1.1: Δομή και λειτουργία ενός νευρώνα

Εικόνα 4.1.2: Γραφική παράσταση βηματικής συνάρτησης

Εικόνα 4.1.3: Γραφική παράσταση σιγμοειδούς συνάρτησης



Εικόνα 4.1.4: Γραφική παράσταση υπερβολικής εφαπτομένης (tanh)

Εικόνα 4.5.1: Τεχνητό Νευρωνικό δίκτυο με κρυφό επίπεδο

Εικόνα 5.1.1: Το Android Studio είναι διαθέσιμο για download στο

<https://developer.android.com/sdk/index.html>

Εικόνα 5.2.1: Android SDK

Εικόνα 5.3.1: Neuroph Framework

Εικόνα 5.4.1: Παράδειγμα απλού νευρώνα

Εικόνα 5.4.2: Δείγμα αρχιτεκτονικής πολυεπίπεδων αισθητήρων

Εικόνα 6.1.1: Μοντέλο αναπαράστασης εικόνας σε μονοδιάστατο πίνακα και νευρωνικού δικτύου

Εικόνα 6.1.2: Πλαίσιο εξερεύνησης των μαθηματικών τιμών των νευρώνων στο NeurophStudio

Εικόνα 6.1.3: Αρχιτεκτονική του νευρωνικού δικτύου

Εικόνα 6.1.4: Πάνελ αρχικοποίησης των συντελεστών του αλγορίθμου

Εικόνα 6.1.5: Αποτελέσματα αναγνώρισης χαρτονομίσματος από το dataset

Εικόνα 6.1.6: Αποτελέσματα αναγνώρισης χαρτονομίσματος εκτός το dataset

Εικόνα 6.2.1: Στιγμιότυπο εφαρμογής

Εικόνα 6.3.1: Φάκελοι και υποφάκελοι εφαρμογής

Εικόνα 6.3.2: AndroidManifest.xml

Εικόνα 6.3.3: Layout εφαρμογής (content_main.xml)

Εικόνα 6.3.4: Κώδικας loadDataRunnable()



Ευρετήριο Πινάκων

Πίνακας 6.1.1: Εικόνες δείγματα του training set

Πίνακας 6.1.5: Αναπαράσταση σφάλματος νευρωνικού δικτύου ως προς τις επαναλήψεις κατά τη διάρκεια της εκπαίδευσης

Πίνακας 7.1: Χαρακτηριστικά συσκευών που χρησιμοποιήθηκαν στις δοκιμές



Βιβλιογραφία

Alexandridis, A., E. Chondrodima, K. Moutzouris, D. Triantis, 2012. "A neural network approach for the prediction of the refractive index based on experimental data", Journal of Materials Science, 47, p. 883-891.

Kumar, B. P. V., P. Venkataram, 2002. "Prediction-based location management using multilayer neural networks", Journal of Indian institute of science, 82, p. 7-20.

Kohonen, T., 1988. "The Neural Phonetic Typewriter", IEEE Computer, 27, p. 11-21.

Hu, W., D. Xie, T. Tan, S. Maybank, 2004. "Learning Activity Patterns Using Fuzzy Self-Organizing Neural Network", IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics, 34, p. 1619-1626.

Heaton, J., 2008. Introduction to Neural Networks with Java, 2, Heaton Research, Inc.

Gurney, K., 2002. An Introduction to Neural Networks, Routledge

Grossberg, S., 1977. "Classical and Instrumental Learning by Neural Networks", in Progress in Theoretical Biology. 3, p.52-140. New York: Academic Press.



Grossberg, S., 1982. Studies of Mind and Brain: Neural Principles of Learning
Perception, Development, Cognition, and Motor Control, Boston:
Reidel
Press.

Rabiner, L. R., 1989. "A Tutorial on Hidden Markov Models and
Selected
Applications in Speech Recognition", IEEE, 77, p.257-285



Σύνδεσμοι

- [1] <http://www.who.int/mediacentre/factsheets/fs282/en/>
- [2] <https://el.wikipedia.org/wiki/Smartphone>
- [3] <http://www.smartinsights.com/mobile-marketing/mobile-marketing-analytics/mobile-marketing-statistics>
- [4] <https://el.wikipedia.org/wiki/Android>
- [5] <http://www.allaboutandroid.gr/?p=6362>
- [6] <http://www.technotification.com/2014/04/interesting-facts-about-android-os.html>
- [7] [http://en.wikipedia.org/wiki/Android_\(operating_system\)](http://en.wikipedia.org/wiki/Android_(operating_system))
- [8] https://en.wikipedia.org/wiki/Android_version_history
- [9] <http://www.edureka.co/blog/beginners-guide-android-architecture/>
- [10] https://el.wikipedia.org/wiki/Android_Studio
- [11] https://en.wikipedia.org/wiki/Android_software_development
- [12] <http://neuroph.sourceforge.net/>
- [13] <http://www.androidust.com/what-are-different-android-versions-released/>
- [14] <https://www.pinterest.com/pin/514888169869182507/>
- [15] <http://news.softpedia.com/news/Android-SDK-Now-Features-Android-2-0-Eclair-Support-125437.shtml>
- [16] <https://www.pinterest.com/pin/452048881318480647/>
- [17] <http://android-developers.blogspot.gr/2010/12/android-23-platform-and-updated-sdk.html>
- [18] <http://www.redmondpie.com/download-and-install-android-3.2-honeycomb-on-asus-eee-pc-or-laptop-how-to-tutorial/>
- [19] <http://www.geek.com/review/android-4-0-ice-cream-sandwich-a-preview-of-the-core-new-features-1432011/>
- [20] <http://vr-zone.com/articles/android-4-3-jelly-bean-running-nexus-devices-caught-on-server-logs/19890.html>
- [21] <http://www.egadgetsinfo.com/2014/12/android-4-444-versions.html>
- [22] <http://www.xda-developers.com/upgrade-to-lollipop/>
- [23] <http://www.pcmag.com/article2/0,2817,2493585,00.asp>



- [24] <http://www.makingdifferent.com/7-must-apps-smartphone/>
- [25] https://en.wikipedia.org/wiki/Computer_vision#Recognition
- [26] https://en.wikipedia.org/wiki/Self-organizing_map
- [27] <http://www.ai-junkie.com/ann/som/som1.html>
- [28] <http://osarena.net/sites/default/files/old-wp/2013/07/neural-Networks.png>
- [29] https://dSPACE.lib.ntua.gr/dSPACE2/bitstream/handle/123456789/6447/daflae_training.pdf?sequence=3
- [30] <http://slideplayer.gr/slide/2951768/>
- [31] <https://upload.wikimedia.org/math/b/3/8/b38f3b868183c6d9a4ae1fe876eae52.png>
- [32] https://upload.wikimedia.org/wikipedia/commons/thumb/d/d9/Dirac_distribution_CDF.svg/325px-Dirac_distribution_CDF.svg.png
- [33] <https://commons.wikimedia.org/wiki/File:Logistic-curve.svg>
- [34] https://el.wikipedia.org/wiki/%CE%A3%CE%B9%CE%B3%CE%BC%CE%BF%CE%B5%CE%B9%CE%B4%CE%AE%CF%82_%CF%83%CF%85%CE%BD%CE%AC%CF%81%CF%84%CE%B7%CF%83%CE%B7
- [35] <http://mathworld.wolfram.com/HyperbolicTangent.html>
- [36] https://el.wikipedia.org/wiki/%CE%9D%CE%B5%CF%85%CF%81%CF%89%CE%BD%CE%B9%CE%BA%CF%8C_%CE%B4%CE%AF%CE%BA%CF%84%CF%85%CE%BF
- [37] http://www.icsd.aegean.gr/lecturers/kavallieratou/PattRec_files/4other_class.pdf
- [38] <http://users.auth.gr/~voyatzis/SeniorThesis/mTsouxnika.pdf>
- [39] http://webcache.googleusercontent.com/search?q=cache:CHL2o5gHB70J:aetos.it.teithe.gr/~gouliana/files/neuronics/MM_ERGASTHRIO_Askhseis_2011.doc+&cd=3&hl=el&ct=clnk&gl=gr



- [40] https://en.wikipedia.org/wiki/Delta_rule
- [41] http://www.mu-sigma.com/analytics/images/cafe_cerebral/Multilayer%20perceptron.gif
- [42] <http://infolab.cs.unipi.gr/theses/Chondrodima.pdf>
- [43] <https://www.makingmoneywithandroid.com/tag/android-sdk/>